

LISTEN.  
THINK.  
SOLVE.<sup>SM</sup>

INTEGRATED PRODUCTION & PERFORMANCE SUITE



Performance & Visibility

# FactoryTalk® View Machine Edition



## USER'S GUIDE VOLUME 1

PUBLICATION VIEWME-UM004E-EN-E—August 2007  
Supersedes Publication VIEWME-UM004D-EN-E

ALLEN-BRADLEY • ROCKWELL SOFTWARE

**Rockwell  
Automation**

<b>Contacting Rockwell</b>	Customer Support Telephone — 1-440-646-3434
	Online support — <a href="http://support.rockwellautomation.com">http://support.rockwellautomation.com</a>
<b>Copyright Notice</b>	© 2007 Rockwell Automation Technologies Inc. All rights reserved. Printed in USA.
	This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies Inc. is strictly prohibited. Please refer to the license agreement for details.
<b>Trademark Notices</b>	Allen-Bradley, ControlLogix, FactoryTalk, PLC-2, PLC-3, PLC-5, Rockwell Automation, Rockwell Software, RSLinx, RSView, the Rockwell Software logo, and VersaView are registered trademarks of Rockwell Automation, Inc.
	The following logos and products are trademarks of Rockwell Automation, Inc.:
	RSMACC, RSLogix, Data Highway Plus, DH+, RSView, FactoryTalk View, RSView Studio, FactoryTalk View Studio, PanelBuilder, PanelView, RSView Machine Edition, RSView ME Station, and WINtelligent.
	FactoryTalk Activation, FactoryTalk Administration Console, FactoryTalk Alarms and Events, FactoryTalk Automation Platform, FactoryTalk Services Platform, FactoryTalk Diagnostics, FactoryTalk Directory, FactoryTalk Live Data, RSAAssetSecurity, and FactoryTalk Security.
	PanelView, RSLinx Classic, RSLinx Enterprise, SLC 5, and SLC 500
<b>Other Trademarks</b>	ActiveX, Microsoft, Microsoft Access, SQL Server, Visual Basic, Visual C++, Visual SourceSafe, Windows, Windows ME, Windows NT, Windows 2000, Windows Server 2003, and Windows XP are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
	Adobe, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.
	ControlNet is a registered trademark of ControlNet International.
	DeviceNet is a trademark of the Open DeviceNet Vendor Association Inc. (ODVA)
	Ethernet is a registered trademark of Digital Equipment Corporation, Intel, and Xerox Corporation.
	OLE for Process Control is a registered trademark of the OPC Foundation.
	Oracle, SQL*Net, and SQL*Plus are registered trademarks of Oracle Corporation.
	All other trademarks are the property of their respective holders and are hereby acknowledged.
<b>Warranty</b>	This product is warranted in accord with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.
	This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at anytime without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

# Contents

## Preface

Product name changes .....	P-1
About the documentation .....	P-2
Finding the information you need .....	P-2
Try the User's Guide and Help first .....	P-2
Information on the Internet .....	P-3
Contacting Rockwell Automation Technical Support .....	P-3

## 1 • Getting Started

The parts of FactoryTalk View Machine Edition .....	1-1
Additional software .....	1-1
FactoryTalk Administration Console .....	1-1
The FactoryTalk View Machine Edition tools .....	1-2
FactoryTalk View Studio tools .....	1-2
FactoryTalk tools .....	1-3
FactoryTalk Activation tools .....	1-3

## 2 • Exploring FactoryTalk View Studio

Starting and exiting FactoryTalk View Studio .....	2-1
Starting FactoryTalk View Studio .....	2-1
Exiting FactoryTalk View Studio .....	2-1
Opening sample applications .....	2-2
Exploring the FactoryTalk View Studio main window .....	2-4
The menu bar .....	2-4
The toolbar .....	2-4
The Explorer window .....	2-5
The workspace .....	2-5
The Diagnostics List .....	2-5
The status bar .....	2-6
Workbook tabs .....	2-7
Showing and hiding items in the main window .....	2-7
Using the Explorer window .....	2-8
Viewing the Explorer window .....	2-10
Moving the Explorer window .....	2-10

Working with editors . . . . .	2-10
Locating editors . . . . .	2-10
Viewing an editor's components . . . . .	2-11
Opening editors . . . . .	2-11
Closing editors . . . . .	2-12
Entering information in spreadsheets . . . . .	2-12
Printing . . . . .	2-13
Selecting a printer . . . . .	2-14
Printing at run time . . . . .	2-15

### **3 • Planning applications**

Understanding the process . . . . .	3-1
Collecting data . . . . .	3-2
Designing an HMI tag database . . . . .	3-2
Collecting information . . . . .	3-2
Organizing tags . . . . .	3-2
Planning graphic displays . . . . .	3-3
Developing a hierarchy of displays . . . . .	3-3
Creating a template to ensure consistency . . . . .	3-3
Designing displays . . . . .	3-4
Usability . . . . .	3-4
Planning languages . . . . .	3-5
Planning alarms . . . . .	3-5
Providing information for the operator . . . . .	3-5
Local and information messages . . . . .	3-5
Diagnostics messages . . . . .	3-6
Planning trends . . . . .	3-6
Planning recipes . . . . .	3-6
Designing a secure system . . . . .	3-7

### **4 • Working with applications**

What is an application? . . . . .	4-1
Application versus project . . . . .	4-1
HMI project file . . . . .	4-2
Runtime application file . . . . .	4-2
Component files . . . . .	4-2
External folders . . . . .	4-2
Default log file locations for PanelView Plus or PanelView Plus CE	
applications . . . . .	4-3
Naming files . . . . .	4-4



Creating, importing, opening, and closing applications.....	4-4
Creating applications .....	4-4
Importing applications .....	4-5
Opening applications .....	4-7
Opening multiple applications .....	4-9
Opening and editing applications from earlier versions of RSView .....	4-9
Closing applications .....	4-10
Renaming, copying, deleting, backing up, and restoring applications.....	4-10
About project settings.....	4-11
Project window size and runtime screen resolution .....	4-11
Title bar .....	4-14
Specifying project settings.....	4-16
Viewing application properties .....	4-17
Viewing HMI server properties .....	4-17

## 5 • Setting up communications

About OPC communications .....	5-1
Summary of steps .....	5-1
About data servers .....	5-2
Creating data servers .....	5-3
Setting up RSLinx Enterprise data servers .....	5-3
Setting up an OPC data server .....	5-4
Updating data server caches .....	5-5

## 6 • Working with tags

Types of tags .....	6-1
Data server tags.....	6-1
HMI tags .....	6-2
The data source .....	6-2
Basic steps for using tags .....	6-2
When to use data server tags .....	6-3
Eliminating duplication.....	6-3
Using complex data .....	6-3
Steps for using data server tags .....	6-3
When to use HMI tags .....	6-4
Scaling, offsetting, or providing a range for data.....	6-4
Storing values in FactoryTalk View memory.....	6-4
Steps for using HMI tags .....	6-5
Browsing for tags .....	6-5
Using the Tag Browser .....	6-6
Showing server names .....	6-7

Browsing for off-line tags .....	6-7
Using tags and expressions in your application .....	6-8
Assigning tags .....	6-9
Assigning tags to graphic objects .....	6-9
Using expressions to manipulate tag values .....	6-11
Substituting tag names used in graphic objects .....	6-11
Logging tag values .....	6-12
Using macros to assign values to tags .....	6-12

## 7 • Using HMI tags

HMI tag types .....	7-1
Analog tags that use floating-point values .....	7-2
How values are rounded .....	7-2
Using the Tags editor .....	7-3
Updating the contents of the Tags editor .....	7-4
Searching for HMI tags .....	7-4
Data sources .....	7-5
The data source .....	7-5
Device .....	7-5
Memory .....	7-5
Addressing syntax for device tags .....	7-5
Organizing HMI tags .....	7-7
Naming tags .....	7-7
Using folders to group tags .....	7-7
Viewing tag statistics .....	7-8
Other methods for creating HMI tags .....	7-8
Creating tags as needed in other FactoryTalk View editors .....	7-8
Creating tags as needed in the Data Log Models editor .....	7-9
Importing tags from a PLC database .....	7-9
Using the Tag Import and Export Wizard .....	7-11

## 8 • Setting up global connections

About global connections .....	8-1
Setting up global connections .....	8-2
Updating the date and time .....	8-2
Changing displays .....	8-4
Controlling display changes remotely .....	8-4
Remote display changes and security .....	8-4
Setting up remote display changes .....	8-4
Printing displays .....	8-5
Running macros .....	8-6

Setting up backlight intensity remotely .....	8-6
Monitoring runtime RAM usage .....	8-7

## 9 • Setting up alarms

About alarms .....	9-1
Multiple language alarm messages.....	9-1
Other multiple language alarm features .....	9-2
Summary of steps .....	9-2
Setting up alarms .....	9-3
Preparing to set up alarms .....	9-3
The data source .....	9-3
Tags and expressions .....	9-3
Identifying alarm conditions.....	9-4
Importing and exporting alarm setup files .....	9-4
How alarms work .....	9-4
Alarm triggers and trigger values .....	9-4
Filtering alarm triggers in multiple languages .....	9-5
Alarm notification methods .....	9-5
Displaying alarm information.....	9-6
Interacting with alarms .....	9-7
The alarm log file .....	9-10
Alarm trigger data types .....	9-11
The Value trigger type .....	9-11
The Bit trigger type.....	9-12
The Least Significant Bit (LSBit) trigger type.....	9-13
Tips for using array tags .....	9-15
Equivalent data types .....	9-16
RSLinx Enterprise tag syntax .....	9-16
KEPServerEnterprise tag syntax.....	9-17
Creating alarm messages in multiple languages.....	9-17
Optional alarm connections .....	9-17
Connections that work with a specific alarm trigger .....	9-18
How the Handshake connection works.....	9-18
How the Ack connection works .....	9-19
How the Remote Ack connection works .....	9-19
How the Remote Ack Handshake connection works .....	9-20
Ensuring alarm messages are read by the data source before sending new messages .....	9-21
Methods of alarm message handshaking .....	9-21
Holding the message for a specific period of time .....	9-21

Holding the message until the data source acknowledges that it has read the message .....	9-22
How messages are queued .....	9-22
How the Message connection works .....	9-22
How the Message Notification connection works .....	9-23
How the Message Handshake connection works .....	9-23
Connections that apply to all alarms .....	9-23
How the Silence connection works .....	9-24
How the Remote Silence connection works .....	9-24
How the Remote Ack All connection works .....	9-24
How the Status Reset connection works .....	9-24
How the Remote Status Reset connection works .....	9-25
How the Close Display connection works .....	9-25
How the Remote Close Display connection works .....	9-25
The [ALARM] display .....	9-25
The [ALARM BANNER] display .....	9-26
The [ALARM MULTI-LINE] display .....	9-27
The [STATUS] display .....	9-28
The [HISTORY] display .....	9-29
Using displays from the library in your application .....	9-30
Creating your own alarm display .....	9-30
Opening and closing the alarm display .....	9-31
Opening the display .....	9-31
Closing the display .....	9-31
How the alarm list graphic object works .....	9-32
What is displayed .....	9-32
How the list scrolls .....	9-33
How the alarm banner graphic object works .....	9-33
What is displayed .....	9-34
How the alarm status list graphic object works .....	9-34
What is displayed .....	9-34
What happens when the display is opened .....	9-35
Using buttons with the alarm history and alarm objects .....	9-35
Alarm buttons .....	9-35
Linking buttons to objects .....	9-37
Key buttons .....	9-37
Using alarm buttons to acknowledge, silence, clear, and delete alarms .....	9-38
Acknowledging the selected alarm .....	9-38
Acknowledging all alarms .....	9-39
Silencing alarms .....	9-39
Clearing and deleting messages .....	9-39
Using alarm buttons to sort alarms and reset alarm status .....	9-40

Sorting alarms . . . . .	9-40
Resetting alarm status . . . . .	9-40
Retaining alarm status . . . . .	9-40
Changing the alarm status displayed in the alarm status list . . . . .	9-40

## 10 • Setting up FactoryTalk Diagnostics

About FactoryTalk Diagnostics . . . . .	10-1
Browsing diagnostics messages . . . . .	10-1
How to set up FactoryTalk Diagnostics . . . . .	10-1
Destinations . . . . .	10-2
Message routing . . . . .	10-3
Categories . . . . .	10-3
Message severities . . . . .	10-3
Audiences . . . . .	10-4
Displaying diagnostics messages during application development . . . . .	10-4
Viewing FactoryTalk Diagnostics log files . . . . .	10-5
Using the Diagnostics Setup tool . . . . .	10-6
Logging to an ODBC database . . . . .	10-7
Routing messages . . . . .	10-8
Receiving messages from a PanelView Plus or PanelView Plus CE terminal . . . . .	10-8
Displaying and printing diagnostics messages at run time . . . . .	10-8
Using the Diagnostics List Setup editor . . . . .	10-9
Setting up how messages are displayed and printed at run time . . . . .	10-10
The [DIAGNOSTICS] display . . . . .	10-10
Creating your own diagnostics display . . . . .	10-11
Opening and closing the diagnostics display . . . . .	10-11
Opening the display . . . . .	10-11
Closing the display . . . . .	10-11
How the diagnostics list graphic object works . . . . .	10-11
What is displayed . . . . .	10-12
Using buttons with the diagnostics list . . . . .	10-12

## 11 • Setting up security

Using security with your application . . . . .	11-1
If you don't want to use FactoryTalk Security . . . . .	11-2
If you don't use FactoryTalk View user accounts . . . . .	11-3
If you use FactoryTalk View user accounts . . . . .	11-3
Working with the Runtime Security editor . . . . .	11-3
The DEFAULT user . . . . .	11-4
Setting up users for 4.00 and later applications . . . . .	11-4

Setting up users for 3.20 and earlier applications.....	11-6
Changing RSView 3.20 and earlier user passwords.....	11-7
Adding 3.20 and earlier users or groups from a Windows domain .....	11-7
Removing 3.20 and earlier users or groups .....	11-8
How user accounts and security codes work .....	11-9
Assigning security to graphic displays.....	11-9
Providing a way for users to log in and log out .....	11-12
Logging in.....	11-12
Logging out.....	11-13
Logging out automatically .....	11-13
Preventing unauthorized users from stopping the application .....	11-14
To prevent unauthorized users from stopping the application .....	11-14
Additional step for personal computers .....	11-14
Additional step for PanelView Plus CE applications .....	11-14
Setting up FactoryTalk Security for your application .....	11-14
Summary of steps .....	11-17
Creating FactoryTalk Security users.....	11-17
Creating FactoryTalk Security user groups .....	11-19
Setting up security access to the FactoryTalk Directory .....	11-19
Setting up security access to the application .....	11-20
Setting up security access to System policies, groups, and users.....	11-20
Setting up security access to networks and devices .....	11-20
Specifying which users can back up and restore FactoryTalk Directory and FactoryTalk View applications .....	11-21
Specifying activities to track for audit purposes.....	11-21
Specifying policies for passwords, accounts, and FactoryTalk sign-on.....	11-21
Summary of security features in a sample startup display .....	11-22

## **12 • Setting up language switching**

About language switching .....	12-1
The default language.....	12-2
Summary of steps .....	12-3
Setting up Windows for language switching .....	12-4
Installing Windows languages .....	12-4
Setting up Windows fonts.....	12-4
Windows locale settings.....	12-4
Adding languages to the application .....	12-5
Removing languages.....	12-5
Exporting application text strings for translation.....	12-6
Exporting text in Unicode format.....	12-6
Exporting text to a Microsoft Excel spreadsheet .....	12-7

Excel spreadsheet file name format .....	12-7
Exported language string file locations .....	12-8
Problems exporting .....	12-8
Translating application text in Excel spreadsheet files .....	12-9
Translating application text in Unicode files .....	12-9
File name and format .....	12-9
Opening the text file in Microsoft Excel .....	12-9
Saving the text file in Microsoft Excel .....	12-10
Differences in file format for files saved in Excel .....	12-10
Saving the Unicode text file in Notepad .....	12-11
File schema .....	12-11
Working with pairs of double quotes .....	12-12
Working with backslashes and new line characters .....	12-12
Importing text .....	12-13
Problems importing .....	12-13
Canceling importing .....	12-14
Setting up multiple language support for graphic libraries .....	12-14
Using graphic libraries that support multiple languages .....	12-15

## 13 • Setting up display navigation

About display navigation .....	13-1
Developing a hierarchy of displays .....	13-1
Testing display navigation .....	13-3
Using graphic objects to navigate .....	13-3
Switching languages .....	13-3
Display type .....	13-4
Goto display buttons .....	13-4
Goto configure mode buttons .....	13-4
Return to display buttons .....	13-5
Close display buttons .....	13-6
Display list selectors .....	13-6
Shutdown buttons .....	13-7
Controlling display changes remotely .....	13-7

## 14 • Creating runtime applications

Specifying startup settings .....	14-1
Testing your application .....	14-2
Creating runtime application files .....	14-2
Creating .mer files for previous versions .....	14-2
Converting .mer files to development applications .....	14-3
Converting runtime application files to development applications .....	14-6

## **15 • Running applications on a personal computer**

Summary of steps .....	15-1
Moving applications to the runtime computer .....	15-3
Starting FactoryTalk View ME Station .....	15-4
Loading and running applications .....	15-4
Shutting down applications .....	15-5
What happens when the application shuts down .....	15-6
Changing application settings .....	15-6
Editing device shortcuts .....	15-6
Looking up contact information for technical support .....	15-7
Setting up FactoryTalk Diagnostics on the runtime computer .....	15-7
Setting up serial ports for use with KEPServerEnterprise .....	15-8
Setting up RSLinx Enterprise communication drivers .....	15-9
Specifying the printers to use at run time .....	15-10
Specifying startup options for FactoryTalk View ME Station .....	15-11
Deleting log files on the runtime computer .....	15-15
Running a newer version of the application .....	15-15
Deleting log files manually .....	15-16
Turning off the FactoryTalk Directory Server warning .....	15-16
Specifying time, date, and number formats .....	15-16
Using the DeskLock tool .....	15-17

## **16 • Transferring applications to a PanelView Plus or PanelView Plus CE terminal**

Summary of steps .....	16-1
Starting FactoryTalk View ME Station .....	16-3
Specifying the driver to use for the transfer .....	16-3
Setting up a driver for the transfer on the development computer .....	16-5
Downloading applications and Windows TrueType fonts .....	16-5
About the download .....	16-5
Serial downloads .....	16-6
Uploading applications from the PanelView Plus or PanelView Plus CE terminal .....	16-6
About the upload .....	16-7
Serial uploads .....	16-7
Comparing applications .....	16-7

**17 • Using your application**

Logging in to a Windows domain . . . . .	17-1
Logging in to the application . . . . .	17-2
4.00 and later applications . . . . .	17-2
3.20 and earlier applications . . . . .	17-2
What happens when a user logs in . . . . .	17-3
Problems with logging in . . . . .	17-4
Changing passwords . . . . .	17-4
Logging out . . . . .	17-6
Entering numeric values . . . . .	17-6
Activating the cursor point . . . . .	17-6
Ramping numeric values . . . . .	17-7
Using the numeric pop-up keypad . . . . .	17-7
Using the numeric pop-up scratchpad . . . . .	17-7
Using buttons and keys with the numeric pop-up windows . . . . .	17-8
How values are ramped . . . . .	17-9
How values are calculated . . . . .	17-9
Problems with the numeric pop-up windows . . . . .	17-10
Entering string values . . . . .	17-10
Using the string pop-up keyboard . . . . .	17-11
Using the string pop-up character input . . . . .	17-11
Using the string pop-up scratchpad . . . . .	17-13
Using buttons and keys with the string pop-up windows . . . . .	17-13
What is written to the Value connection . . . . .	17-14
Problems with the string pop-up windows . . . . .	17-15
Changing tag values . . . . .	17-15
Viewing tag data . . . . .	17-17
Viewing alarms and messages . . . . .	17-19
Viewing information about runtime communication errors . . . . .	17-20
Changing languages . . . . .	17-20

**18 • Working with components**

Editors that have components . . . . .	18-1
Working with components . . . . .	18-1
Creating components . . . . .	18-2
Opening components . . . . .	18-2
Saving components . . . . .	18-2
Closing components . . . . .	18-2
Adding components into an application . . . . .	18-3
Deleting components . . . . .	18-4
Removing components . . . . .	18-4

Renaming components .....	18-4
Duplicating components .....	18-5
Printing .....	18-5

## 19 • Using graphic displays

About graphic displays and graphic objects .....	19-1
Before you begin .....	19-2
Using the Graphics editor .....	19-2
Creating and opening graphic displays .....	19-3
Importing and exporting graphic displays .....	19-4
Tools and tips for working in the Graphics editor .....	19-5
Using context menus .....	19-5
Using the toolbars .....	19-7
Showing displays in grayscale .....	19-8
Using the grid .....	19-8
Zooming in and out .....	19-9
Correcting mistakes .....	19-10
Testing your displays as you work .....	19-10
Setting up graphic displays .....	19-11
Specifying display settings .....	19-11
About display types .....	19-12
Resizing displays .....	19-13
Creating a background for your display .....	19-14
Using graphic libraries .....	19-14
Creating graphic libraries .....	19-15
Using libraries as displays in your application .....	19-16
Using libraries to store displays with multiple languages .....	19-17
Location of library components .....	19-19
Importing images into your application .....	19-20
Bitmap images that come with FactoryTalk View Studio .....	19-20
Importing bitmap and JPEG images .....	19-21
Using the Image Browser to import images .....	19-23
Tips for using images .....	19-24
Using local messages .....	19-25
Local messages versus information messages .....	19-25
Summary of steps .....	19-26
Using the Local Messages editor .....	19-26
Preparing to set up local messages .....	19-26
How local messages work .....	19-27
Local messages and trigger values .....	19-28
Creating local messages in multiple languages .....	19-29

How the local message display graphic object works .....	19-29
Printing displays .....	19-29
Printing displays at run time .....	19-30

## 20 • Using graphic objects

Types of graphic objects .....	20-1
About connections .....	20-2
Illustrating your displays .....	20-2
Controlling the application .....	20-4
Starting and controlling processes .....	20-4
Displaying processes and values graphically .....	20-6
Working with lists, trends, alarm banners, and numeric input objects .....	20-7
Entering and displaying numeric and string values .....	20-9
Displaying alarms and messages .....	20-10
Selecting tools for creating graphic objects .....	20-10
Before you begin creating objects .....	20-12
Creating graphic objects .....	20-12
Creating drawing objects .....	20-12
Creating text .....	20-13
Creating images .....	20-14
Creating panels .....	20-16
Creating arcs and wedges .....	20-16
Creating ellipses and circles .....	20-17
Creating freehand shapes .....	20-17
Creating lines .....	20-18
Creating polygons and polylines .....	20-18
Creating rectangles and squares .....	20-19
Creating rounded rectangles and squares .....	20-20
Using .wmf and .dxf files .....	20-20
Using ActiveX objects .....	20-21
Tools and tips for working with objects .....	20-23
Selecting and deselecting objects .....	20-23
Using the Object Explorer .....	20-24
Highlighting objects in the Object Explorer .....	20-25
Using the Properties dialog box .....	20-26
Using the Property Panel .....	20-29
Setting up properties .....	20-30
Assigning tags and expressions to an object's connections .....	20-31
Coloring objects using the color toolbars .....	20-32
Naming objects .....	20-34
Testing how objects look in different states .....	20-35

Assigning tags and expressions to graphic objects .....	20-36
Assigning tags .....	20-36
Using expressions to manipulate tag values .....	20-38
Replacing tags using tag substitution .....	20-38
Using tag placeholders .....	20-39
Performing basic operations on objects .....	20-40
Moving objects .....	20-41
Copying objects .....	20-42
Duplicating objects .....	20-43
Resizing objects .....	20-44
Reshaping drawing objects .....	20-45
Deleting objects .....	20-46
Working with groups of objects .....	20-47
Grouping and ungrouping objects .....	20-47
Editing groups of objects .....	20-48
Editing objects within a group .....	20-48
Arranging objects .....	20-49
Layering objects .....	20-49
Aligning objects .....	20-50
Spacing objects .....	20-53
Flipping drawing objects .....	20-54
Rotating drawing objects .....	20-55
Locking objects into position .....	20-56

## 21 • Setting up graphic objects

Setting up objects' spatial properties, names, and visibility .....	21-1
Tips for setting up objects with states .....	21-2
Copying and pasting properties from one state to another .....	21-3
Adding and removing states .....	21-3
Setting up how objects are used at run time .....	21-3
Positioning objects for touch screens .....	21-4
Using touch margins .....	21-4
Assigning function keys to buttons .....	21-5
Function key equivalents .....	21-6
Using the keyboard to navigate to and select objects .....	21-7
What input focus looks like .....	21-7
Using the keys on the keyboard or keypad .....	21-8
Removing objects from and adding objects to the tab sequence .....	21-8
Linking buttons to objects .....	21-9
Repeating a button's action by holding down the button .....	21-12

Ensuring values are read by the data source before sending new values . . . . .	21-13
Methods of Enter key handshaking . . . . .	21-13
Holding the value for a specific period of time . . . . .	21-14
Holding the value until the data source acknowledges that it has read the value . . . . .	21-14
Time, date, and number formats for graphic objects . . . . .	21-16
Setting up buttons . . . . .	21-16
Buttons described later in the chapter . . . . .	21-19
How to use push buttons . . . . .	21-19
Setting up momentary push buttons . . . . .	21-20
Setting up maintained push buttons . . . . .	21-21
Setting up latched push buttons . . . . .	21-23
Setting up multistate push buttons . . . . .	21-24
Setting up interlocked push buttons . . . . .	21-26
Setting up ramp buttons . . . . .	21-27
Setting up numeric displays . . . . .	21-28
How values are displayed . . . . .	21-29
Problems with displaying values . . . . .	21-29
Setting up numeric input enable buttons . . . . .	21-30
Setting up numeric input cursor points . . . . .	21-32
Setting up string displays . . . . .	21-33
How values are displayed . . . . .	21-33
Setting up string input enable buttons . . . . .	21-34
Setting up goto display buttons . . . . .	21-35
Setting up close display buttons . . . . .	21-36
Setting up display list selectors . . . . .	21-37
How to use indicators . . . . .	21-38
Setting up multistate indicators . . . . .	21-39
Setting up symbols . . . . .	21-40
Setting up list indicators . . . . .	21-42
How to use bar graphs, gauges, and scales . . . . .	21-43
Bar graphs make it easy to compare values . . . . .	21-43
Thresholds change a bar graph's fill color . . . . .	21-43
Use bar graphs with scales to show limits . . . . .	21-44
Gauges make it easy to see limits . . . . .	21-44
Thresholds change a gauge's fill color . . . . .	21-44
Setting up bar graphs . . . . .	21-45
Setting up gauges . . . . .	21-46
Setting up scales . . . . .	21-47
Setting up control list selectors . . . . .	21-47
Using buttons with the control list selector . . . . .	21-48
How Enter key handshaking works . . . . .	21-48

Setting up piloted control list selectors .....	21-50
Choosing between piloted control list selectors and control list selectors .....	21-50
How piloted control list selectors work at run time .....	21-50
Setting up local message displays .....	21-53
Setting up macro buttons .....	21-54
Setting up time and date displays .....	21-55
Setting up print alarm history buttons .....	21-56
Setting up print alarm status buttons .....	21-57
Setting up alarm lists .....	21-58
Setting up alarm banners .....	21-59
Setting up alarm status lists .....	21-60
Setting up diagnostics lists .....	21-62
Setting up information message displays .....	21-63

## 22 • Animating graphic objects

Types of animation .....	22-1
Which objects can have which types of animation? .....	22-2
Using the Animation dialog box .....	22-2
About the Animation dialog box .....	22-4
Using Object Smart Path to visually set animation .....	22-4
Testing animation .....	22-5
Using tag names and tag placeholders .....	22-5
Using expressions .....	22-6
Setting minimum and maximum values .....	22-6
Defining a range of motion .....	22-6
Animation that does not use a range of motion .....	22-7
Using Object Smart Path .....	22-7
Setting up the different types of animation .....	22-8
Setting up visibility animation .....	22-8
Setting up color animation .....	22-9
Setting up fill animation .....	22-12
Setting up horizontal position animation .....	22-12
Setting up vertical position animation .....	22-12
Setting up width animation .....	22-13
Setting up height animation .....	22-13
Setting up rotation animation .....	22-13
Setting up horizontal slider animation .....	22-13
Setting up vertical slider animation .....	22-14
Applying animation to groups .....	22-14
Checking the animation on objects .....	22-15
Copying or duplicating objects with animation .....	22-16

Copying animation without copying objects .....	22-16
Setting up animation for global objects .....	22-17

## 23 • Using expressions

About expressions .....	23-1
Expressions that result in floating-point values .....	23-1
Expression components .....	23-2
Using the Expression editor .....	23-2
Using the Expression editor versus typing expressions directly .....	23-2
About the Expression editor .....	23-4
Where you can use expressions .....	23-4
Formatting expressions .....	23-5
Using tag names and tag placeholders .....	23-5
Using tag placeholders instead of tag names .....	23-6
Constants .....	23-6
Arithmetic operators .....	23-7
String operands .....	23-7
Relational operators .....	23-8
How string operands are evaluated .....	23-8
Logical operators .....	23-8
Bitwise operators .....	23-9
Using the left shift operator .....	23-10
Evaluation order of operators .....	23-11
Mathematical functions .....	23-14
Security functions .....	23-15
Language function .....	23-15
If-then-else .....	23-17
Nested if-then-else .....	23-18
Using write expressions .....	23-19

## 24 • Using embedded variables

About embedded variables .....	24-1
Where you can create embedded variables .....	24-2
Creating embedded variables .....	24-2
Embedded variable syntax .....	24-3
Numeric embedded variable syntax .....	24-3
String embedded variable syntax .....	24-4
Time and date embedded variable syntax .....	24-6
How embedded variables are updated at run time .....	24-6

How embedded variables are displayed at run time .....	24-7
Numeric embedded variables .....	24-7
String embedded variables .....	24-8
Time and date embedded variables .....	24-8

## **25 • Using parameters and global objects**

Using tag placeholders and parameter files .....	25-1
Summary of steps .....	25-2
Creating parameter files .....	25-3
Using the Parameters editor .....	25-3
Using global objects .....	25-6
Summary of steps .....	25-7
Creating global object displays and base objects .....	25-7
Creating reference objects .....	25-8
Setting up reference objects' link properties .....	25-9
Deleting the base object .....	25-10
Using global object parameters .....	25-10
Difference between global object parameters and regular parameters .....	25-11
Using global object parameters with group objects .....	25-11
Summary of steps .....	25-12
Using process faceplates .....	25-13

## **26 • Setting up data logging**

Summary of steps .....	26-1
Data log files .....	26-1
Data Log Models .....	26-2
Creating Data Log Models .....	26-3
Data storage locations .....	26-3
Data logging methods .....	26-4
Tags in the data log model .....	26-5
Changing the data log model used at run time .....	26-5
Displaying data logs using the trend graphic object .....	26-6
Problems with data logging .....	26-6

## **27 • Using information messages**

About information messages .....	27-1
Information messages versus local messages .....	27-1
Summary of steps .....	27-1
Using the Information Messages editor .....	27-2
Setting up how information messages are displayed .....	27-3

Preparing to set up information messages .....	27-3
The data source .....	27-3
Identifying the tags and values to monitor .....	27-3
How information messages work .....	27-4
Information messages and trigger values .....	27-5
Creating information messages in multiple languages .....	27-5
The [INFORMATION] display .....	27-6
Creating your own information message display .....	27-7
Opening and closing the information message display .....	27-7
Opening the display .....	27-7
Closing the display .....	27-7
How the information message display graphic object works .....	27-7
What is displayed .....	27-8
Changing the message file used at run time .....	27-8

## 28 • Setting up trends

About trends .....	28-1
Current versus historical data .....	28-2
Time, date, and number formats .....	28-3
Summary of steps .....	28-3
Creating trend objects .....	28-3
Setting up trends .....	28-4
The parts of a trend .....	28-5
Trend border .....	28-5
Trend window .....	28-5
Chart .....	28-6
Y-axis .....	28-6
Vertical axis labels .....	28-6
X-axis .....	28-6
Horizontal axis labels .....	28-6
Pens .....	28-6
Pen icons .....	28-7
Pen markers .....	28-7
Chart types .....	28-7
Standard vs. XY Plots .....	28-7
Isolated graphing .....	28-8
Plotting a value across the full width of the chart .....	28-8
Choosing trend colors, fonts, lines, and markers .....	28-9
The trend border color .....	28-9
The trend window color .....	28-9
Testing the trend .....	28-10

Using the Trends graphic library . . . . .	28-10
Using buttons to control the trend at run time . . . . .	28-11
Printing trend data . . . . .	28-12
Improving clarity of the trend printout . . . . .	28-12
Runtime errors for the trend . . . . .	28-12

## 29 • Setting up RecipePlus

About recipes . . . . .	29-1
Summary of steps . . . . .	29-2
How the recipe system works . . . . .	29-2
RecipePlus selector . . . . .	29-2
RecipePlus table . . . . .	29-2
RecipePlus button . . . . .	29-3
Number format . . . . .	29-4
Numeric limits . . . . .	29-4
Specifying the runtime file location . . . . .	29-5
Storing files outside the HMI project . . . . .	29-5
Storing recipe files with the HMI project . . . . .	29-5
Setting up recipe files . . . . .	29-6
Comparing recipes . . . . .	29-7
Time and date formats . . . . .	29-9
Printing recipes . . . . .	29-9
Creating RecipePlus objects . . . . .	29-9
Testing RecipePlus objects . . . . .	29-10
Using the RecipePlus_Components graphic library . . . . .	29-10
Using buttons with recipe objects . . . . .	29-11
Viewing data values that are saved at run time . . . . .	29-13

## 30 • Using macros

Using macros to assign values to tags . . . . .	30-1
Using the Macros editor . . . . .	30-2
When to use macros . . . . .	30-2
Running macros when tags or expressions change value . . . . .	30-3
Where to assign macros . . . . .	30-4

## APPENDICES

### A • Converting PanelBuilder 1400e applications

Terminology .....	A-1
Summary of steps .....	A-2
Converting PanelBuilder 1400e application files .....	A-3
Equivalent graphic objects .....	A-5
Unsupported graphic objects .....	A-8
Unsupported settings and controls .....	A-8
Controls for transferring runtime application files .....	A-8
Settings and controls for alarms .....	A-8
Invalid characters in screen names .....	A-9
Screen security settings .....	A-9
Block tags .....	A-9
Converting non-RIO communications .....	A-9
Unsupported tag data types .....	A-9
Unsupported initial values .....	A-10
Converting RIO communications .....	A-10
Unsupported PanelBuilder 1400e RIO tags .....	A-11
Unsupported graphic object features .....	A-12
Converting expressions .....	A-13
Equivalent expression syntax .....	A-14
Unsupported expression syntax .....	A-15
Order of precedence .....	A-16

### B • Converting PanelBuilder and PanelBuilder32 applications

Terminology .....	B-1
Summary of steps .....	B-2
Converting PanelBuilder application files .....	B-2
Equivalent graphic objects .....	B-5
Unsupported graphic objects .....	B-6
Unsupported settings and controls .....	B-7
Settings and controls for alarms .....	B-7
Invalid characters in screen names and tag names .....	B-7
Time and date .....	B-7
External fonts .....	B-7
Screen security settings .....	B-7
Power-up options .....	B-7

Converting non-RIO communications .....	B-8
Tags for unsupported communication protocols .....	B-8
Bit array tags .....	B-8
Converting RIO communications .....	B-9
Unsupported PanelBuilder RIO tags .....	B-9
Unsupported graphic object features .....	B-10

## **C • System tags**

Alarms .....	C-1
Graphics .....	C-1
Time .....	C-2
User .....	C-3

## **D • ODBC database schema**

FactoryTalk Diagnostics log table .....	D-1
---	-----

## **E • Importing and exporting alarm XML files**

About XML .....	E-1
Creating alarm XML files by exporting .....	E-1
Editing XML files .....	E-2
Saving XML files in Notepad .....	E-2
Testing XML files .....	E-2
Importing XML files .....	E-2
Error log file .....	E-3
Importing alarm XML files .....	E-3
Alarm setup XML file structure .....	E-3

## **F • RFC1766 names**

Mapping languages to RFC1766 names .....	F-1
--	-----

## **G • Features supported in different versions of FactoryTalk View**

Which versions are supported .....	G-1
Creating runtime application files for previous versions .....	G-1
Features that are not supported in version 4.00 or earlier .....	G-2
Features that are not supported in version 3.20 or earlier .....	G-3
Features that are not supported in version 3.10 or earlier .....	G-4
Features that are not supported in version 3.00 .....	G-5

## H • Importing and exporting graphics XML files

About XML . . . . .	H-1
Creating graphics XML files by exporting. . . . .	H-1
Editing XML files . . . . .	H-2
Saving XML files in Notepad. . . . .	H-2
Testing XML files. . . . .	H-2
Importing XML files . . . . .	H-3
Error log file . . . . .	H-3
Importing graphics XML files . . . . .	H-3
Graphics XML file structure . . . . .	H-3

## Index





FactoryTalk® View Machine Edition is a member of the FactoryTalk View family of products. It is an integrated package for developing and running automation applications.

Designed for use with Microsoft® Windows® 2000, Windows XP, Windows Server 2003 R2, and PanelView™ Plus and PanelView Plus CE terminals running Microsoft Windows CE 4.1, FactoryTalk View Machine Edition gives you all the tools you need to create effective machine-level monitoring and control applications.

## Product name changes

Version 5.00 of FactoryTalk View Machine Edition incorporates a number of product name changes that reflect Rockwell Automation's systems-oriented software offerings. Systems-oriented products are part of the company's Integrated Architecture solution. Here is an overview of some of the name changes you might need to be aware of:

<b>This product</b>	<b>Has been renamed</b>
RSView®, RSView® Enterprise	FactoryTalk® View
RSView® Supervisory Edition™	FactoryTalk® View Site Edition
RSView® SE Distributed™	FactoryTalk® View SE (Network)
RSView® SE Client™	FactoryTalk® View SE Client
RSView® SE Stand-alone™	FactoryTalk® View SE (Local)
RSView® Machine Edition™	FactoryTalk® View Machine Edition
RSView® ME Station™	FactoryTalk® View ME Station
RSView® Studio™	FactoryTalk® View Studio
RSAssetSecurity™	FactoryTalk® Security™
FactoryTalk® Automation Platform™	FactoryTalk® Services Platform™
ControlLogix®	Logix5000
VersaView® CE	PanelView™ Plus CE

## About the documentation

The FactoryTalk View Machine Edition documentation set includes:

- *Release Notes*: Information to read before you begin installing or working with the software.

The Release Notes are available on the FactoryTalk View Machine Edition CD, as well as from the Help menu in FactoryTalk View Studio.

- *FactoryTalk View Machine Edition Installation Guide*: A guide to installing and activating the various components of FactoryTalk View Machine Edition.

The Installation Guide is available on the FactoryTalk View Machine Edition CD, as well as from the Help menu in FactoryTalk View Studio.

- *FactoryTalk View Machine Edition User's Guide, Volume 1* and *FactoryTalk View Machine Edition User's Guide, Volume 2*: Comprehensive information about FactoryTalk View Machine Edition, procedures for creating and running an automation application, and reference information.

For ease of printing, the User's Guide is divided into two parts, available from the Help menu in FactoryTalk View Studio. Volume 1 contains information about setting up and using applications and application-wide functions such as security and alarms. Volume 2 contains information about setting up application components such as graphic displays and data log files. For detailed information about the contents of each guide, see the Bookmarks tab in the online guides.

- *Help*: Online procedures and reference information.

Help is available from the Help menu in FactoryTalk View Studio.

The FactoryTalk View Machine Edition CD also includes this manual:

- *PanelView Plus Terminals User Manual*

## Finding the information you need

You have many options for finding information about how to use FactoryTalk View, or how to solve problems with FactoryTalk View.

### Try the User's Guide and Help first

The User's Guide and Help provide comprehensive information about typical uses of FactoryTalk View. Chances are, your question is answered in the documentation.

To find the answer, use the table of contents and the index in the User's Guide and Help.

You can also perform a full-text search on both the Help and the User's Guide. For information about using Help, see Chapter 5, "Getting the information you need," in the *FactoryTalk View Machine Edition Installation Guide*.

For information about performing searches on the User's Guide, see Adobe® Reader® Help.

## **Information on the Internet**

If you can't find the answer to your question or problem in the User's Guide or Help, you can find information on the World Wide Web.

You can connect to the Rockwell Automation® web site from within FactoryTalk View Studio. To do so, you must have a web browser installed on your computer, and you must have an active Internet connection.

### **To connect to web sites from FactoryTalk View Studio**

1. On the Help menu, select Rockwell Automation on the Web, and then click the name of the web page you want to view.

### **The Rockwell Automation Knowledgebase**

The Knowledgebase web page leads to a comprehensive, searchable database of support information for all Rockwell Automation products.

## **Contacting Rockwell Automation Technical Support**

If you can't find the answer to your question using any of the resources suggested above, contact Rockwell Automation Technical Support at:

Telephone: 440-646-3434

World Wide Web: <http://support.rockwellautomation.com>

Support staff are available Monday to Friday from 8 AM to 7 PM Eastern Standard Time (North America only), except on statutory holidays.

### **When you call**

When you call, be at your computer and be ready to give:

- the product serial number.

You'll find this number on the Activation disk label and in the About FactoryTalk View Studio dialog box available from the Help menu in FactoryTalk View.

- the product version number.
- the type of hardware you are using.

- the exact wording of any messages that appeared on your screen.
- a description of what happened and what you were doing when the problem occurred.
- a description of how you tried to solve the problem.

You may also be required to provide information about the FactoryTalk View add-ons and updates that are installed on your computer.

#### **To view the list of installed add-ons and updates**

1. In FactoryTalk View Studio, click Help, and then click About FactoryTalk View Studio.
2. To view the list of installed add-ons, click Add-Ons.
3. To view the list of installed updates, click Updates.



# Getting Started

FactoryTalk® View Machine Edition is software for developing and running human-machine interface applications. FactoryTalk View Machine Edition is designed for monitoring and controlling automated processes and machines.

For information about installing FactoryTalk View Machine Edition, see the *FactoryTalk View Machine Edition Installation Guide*.

## The parts of FactoryTalk View Machine Edition

FactoryTalk View Machine Edition includes two products:

**FactoryTalk View Studio** is configuration software for developing machine-level applications. This software runs on the Windows® 2000, Windows XP, and Windows Server 2003 R2 operating systems.

**FactoryTalk View ME Station** is a stand-alone runtime environment for machine-level applications. Use FactoryTalk View ME Station to run the applications you develop in FactoryTalk View Studio.

You can run machine-level applications on PanelView™ Plus terminals and PanelView Plus CE terminals with the Windows CE 4.1 operating system, and on personal computers.

## Additional software

The FactoryTalk View Machine Edition CD also includes three additional software products:

**RSLinx® Enterprise™** is a communication server built around FactoryTalk® technology to assist in developing and running your FactoryTalk View ME applications. It is an OPC® 2.05 compliant server and will run on multiple platforms, ranging from PanelView Plus dedicated terminals to personal computers.

**RSLinx® Classic™** is software that provides communications to a wide range of communication drivers. RSLinx Classic is an OPC-compliant data server. RSLinx Classic is available for personal computers only.

**Adobe® Reader® 8.0** is software for reading the online *FactoryTalk View Machine Edition User's Guide*. If desired, you can also use the software to print the User's Guide.

## FactoryTalk Administration Console

The FactoryTalk Services Platform includes software called the FactoryTalk Administration Console. This software allows you to set up FactoryTalk Security™.

FactoryTalk Security is a method for setting up security for users and computers for multiple Rockwell Automation products in one place.

FactoryTalk Security is intended to improve the security of your automation system by limiting access to those with a legitimate need. FactoryTalk Security authenticates user identities and authorizes user requests to access a FactoryTalk-enabled system. These security services are fully integrated into the FactoryTalk Directory and are included as part of the FactoryTalk Services Platform that installs with many products.

You can also set up FactoryTalk Security in FactoryTalk View Studio. For more information, see Chapter 11.

### To open the FactoryTalk Administration Console

1. On the Windows Start menu, select Programs, Rockwell Software, and then click FactoryTalk Administration Console.

For information about using the FactoryTalk Administration Console, see FactoryTalk Help.

## The FactoryTalk View Machine Edition tools

### FactoryTalk View Studio tools

These tools are installed with FactoryTalk View Studio:

- **Application Manager** is software for renaming, copying, deleting, backing up, and restoring applications.
- **DeskLock** is software that prevents Windows 2000, Windows XP, and Windows Server 2003 R2 users from exiting the FactoryTalk View application at run time. You can use this tool to set up a customized desktop and to prevent access to Windows functions. You cannot use DeskLock if you are running your application on a PanelView Plus or PanelView Plus CE terminal.
- **ME Firmware Upgrade Wizard** is software for installing new firmware releases on an ME terminal.
- **ME Transfer Utility** is software for moving a runtime machine-level application (.mer file) from the development computer to PanelView Plus or PanelView Plus CE terminals.
- **Tag Import and Export Wizard** is software for importing or exporting the FactoryTalk View tag database.
- **Diagnostics Viewer** allows you to view FactoryTalk Diagnostics messages using the Windows Event Viewer.

### **To start the tools**

1. On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk View, Tools, and then click the tool to start.

### **To start the Diagnostics Viewer**

1. On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk Tools, and then click Diagnostics Viewer.

These tools are also available from the Tools menu in FactoryTalk View Studio.

### **FactoryTalk tools**

FactoryTalk tools are installed when you install FactoryTalk Services Platform. They are available from the Windows Start menu.

### **To start FactoryTalk tools**

1. On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk Tools, and then click the tool to start.

### **FactoryTalk Activation tools**

FactoryTalk Activation tools are installed when you install FactoryTalk View, if you choose to install the FactoryTalk Activation Server. They are available from the Windows Start menu.

### **To start FactoryTalk Activation tools**

1. On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk Activation, and then click the tool to start.





# Exploring FactoryTalk View Studio

This chapter describes:

- starting and exiting FactoryTalk® View Studio.
- opening sample applications.
- exploring the FactoryTalk View Studio main window.
- using the Explorer window.
- working with editors.
- entering information in spreadsheets.
- printing information in editors.

For information about working with editors' components, see Chapter 18 in Volume 2 of the *FactoryTalk View Machine Edition User's Guide*.

## Starting and exiting FactoryTalk View Studio

FactoryTalk View Studio is configuration software for developing machine-level applications. This software runs on the Microsoft® Windows® 2000, Windows XP, and Windows Server 2003 R2 operating systems.

### Starting FactoryTalk View Studio

#### To start FactoryTalk View Studio

1. On the Windows Start menu, select Programs, and then click FactoryTalk View Studio.

FactoryTalk View comes with several sample applications. We suggest you open the Malthouse sample application now to use while you try out the instructions in the remainder of this chapter. See the instructions on page 2-2.

### Exiting FactoryTalk View Studio

#### To exit FactoryTalk View Studio

1. On the File menu, click Exit.

If there are any unsaved changes in open editors, FactoryTalk View Studio asks you whether to save the changes before exiting the program.

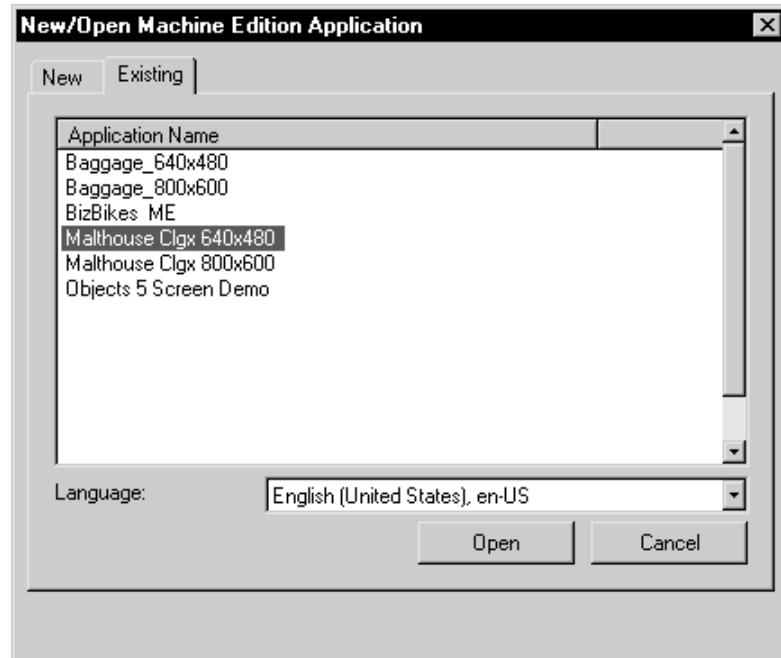
## Opening sample applications

The sample applications are designed for two screen resolutions: 640 x 480 and 800 x 600. Select the application that matches the screen resolution of your runtime monitor.

When you first install FactoryTalk View, sample applications are accessible to all users. For information about setting up users and granting them access to applications, see Chapter 11.

### To open the Malthouse sample application

1. Start FactoryTalk View Studio.
2. Click the Existing tab.



3. Click Malthouse Clgx 640x480 or Malthouse Clgx 800x600.
4. Specify a language for the application, and then click Open.

FactoryTalk View Studio displays the sample application in the Explorer window.



## Problems opening applications

If you have trouble opening an application in FactoryTalk View Studio, the problem could be a Microsoft Windows security setting. You must have read and write access to the folder that contains the HMI projects. For example, if you belong to the Windows user group called Power Users, and the group does not have write access to the HMI projects folder, you won't be able to open applications in FactoryTalk View.

The default location for the HMI projects folder is:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\ME  
(Windows 2000)

or

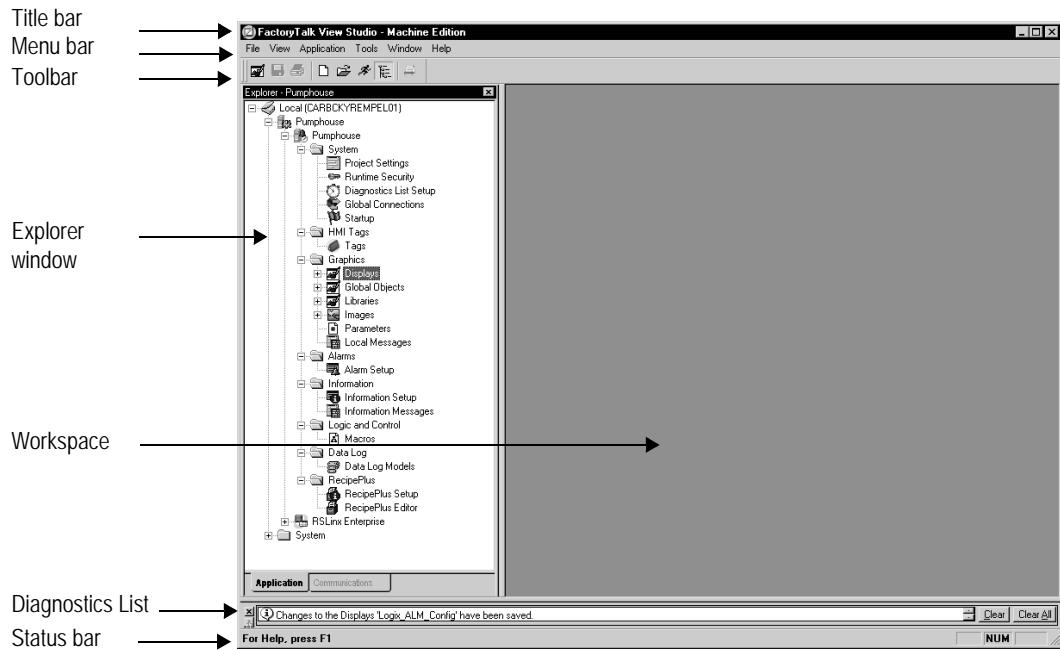
C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\ME  
(Windows XP or Windows Server 2003 R2)

For more information about security, see Chapter 11.

### To set up write access for Power Users

1. Right-click the HMI projects folder, and then click Properties.
2. In the Security tab of the Properties dialog box, select Power Users from the list of groups and user names.
3. In the Permissions box below the list, select Full Control, and then select Allow.

## Exploring the FactoryTalk View Studio main window

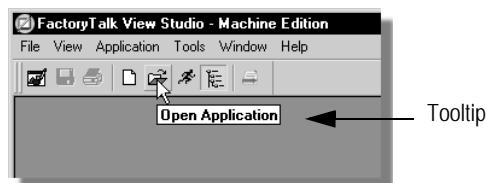


### The menu bar

The menu bar contains the menu items for the active window. Each editor has its own set of menus.

### The toolbar

The toolbar contains buttons for commonly used menu items so you can have quick access to the items without opening the menu. When you point to a button on the toolbar, the button name is displayed in a tooltip and in the status bar.



All editors use the Standard toolbar. The Graphic Displays, Graphic Libraries, Runtime Security, and Tags editors have additional toolbars.

## The Explorer window

The Explorer window has two tabs: the Application tab, and the Communications tab.

The Application tab contains the editors for creating and editing your application. It also contains the System folder for setting up FactoryTalk Security™.

The Communications tab contains the communications tree for the development computer. For more information about using the communications tree, see Help for RSLinx® Enterprise™.

For more information about the Explorer window, see page 2-8. For information about setting up security, see Chapter 11.

## The workspace

The workspace is the blank area of the FactoryTalk View Studio window. You can drag icons from the Explorer window to the workspace to open editors and components. For more information, see page 2-8.

## The Diagnostics List

The Diagnostics List shows messages about system activities. You can specify the types of messages to display in the Diagnostics List, move the list, resize it, and clear the messages in it.

For information about testing graphic displays, see page 19-10.

## Moving the Diagnostics List

You can detach (undock) the Diagnostics List from the main window, and then move the list anywhere on the screen.

To undock the Diagnostics List, drag the grab bars at the bottom left of the Diagnostics List. If you can't see the grab bars, drag the top edge of the Diagnostics List to make it a bit larger.

To detach the  
Diagnostics List, drag  
the grab bars.



To prevent the Diagnostics List from docking automatically while you move it across the screen, hold down the Ctrl key on the keyboard while you move the Diagnostics List.

When the Diagnostics List is undocked, you can make it any size you want, for example to view more than one message at a time. To resize the bar, drag an edge or corner until the bar is the size you want.

To re-attach the Diagnostics List, double-click its title bar.

To clear the selected message in the Diagnostics List, click Clear. To clear all the messages in the Diagnostics List, click Clear All.

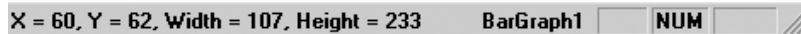
### **Messages in the Diagnostics List**

The types of messages that appear in the Diagnostics List depend on how you set up FactoryTalk Diagnostics on the development system. For information about specifying the types of messages to display, see page 10-3.



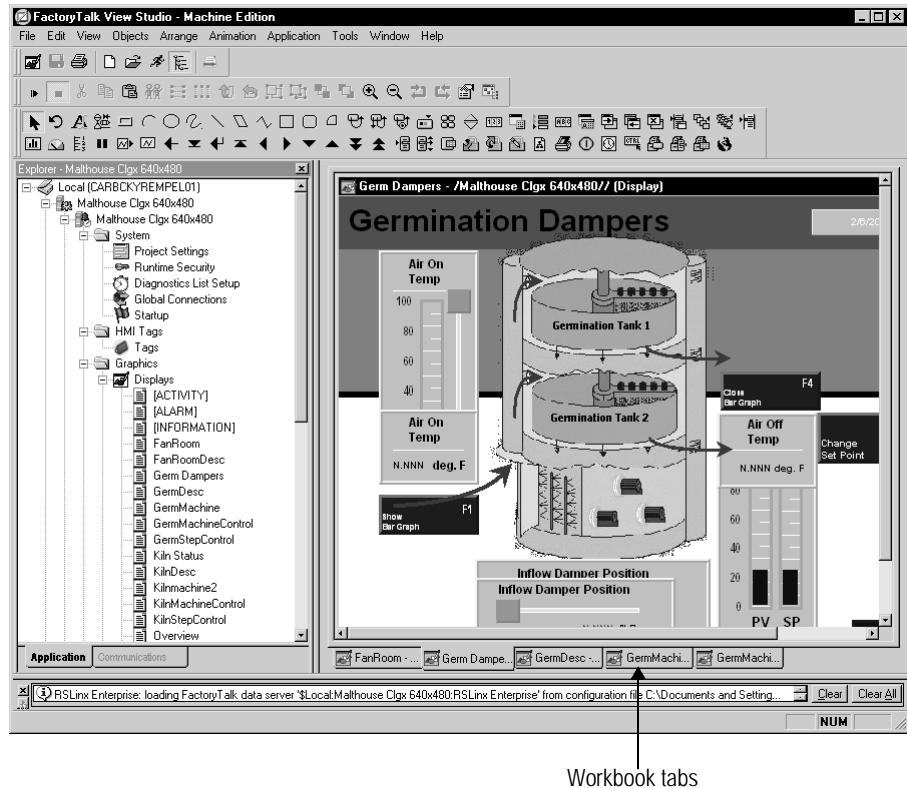
### **The status bar**

The status bar displays information about the active window or about the selected tool or menu item. The information that is displayed depends on where the mouse pointer is. For example, when you select a graphic object in the Graphics editor, the status bar displays information about the selected object.



## Workbook tabs

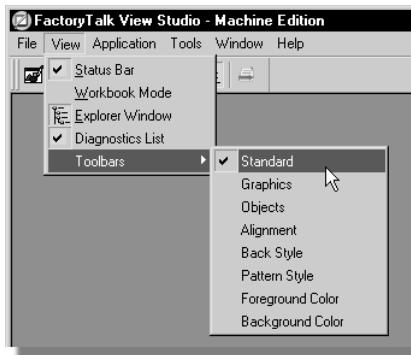
If you select Workbook Mode on the View menu, the workspace displays tabs at the bottom for each open editor or component. Workbook Mode allows you to bring forward an open item more quickly than by using the Window menu.



## Showing and hiding items in the main window

You can display or hide all the items in the main window (except the menu bar) by clicking the items on the View menu.

The status bar, Explorer window, Diagnostics List, and Standard toolbar are visible. Workbook Mode is turned off.



### To hide or display the Explorer window

1. On the View menu, click Explorer Window, or click the Explorer Window tool on the toolbar.

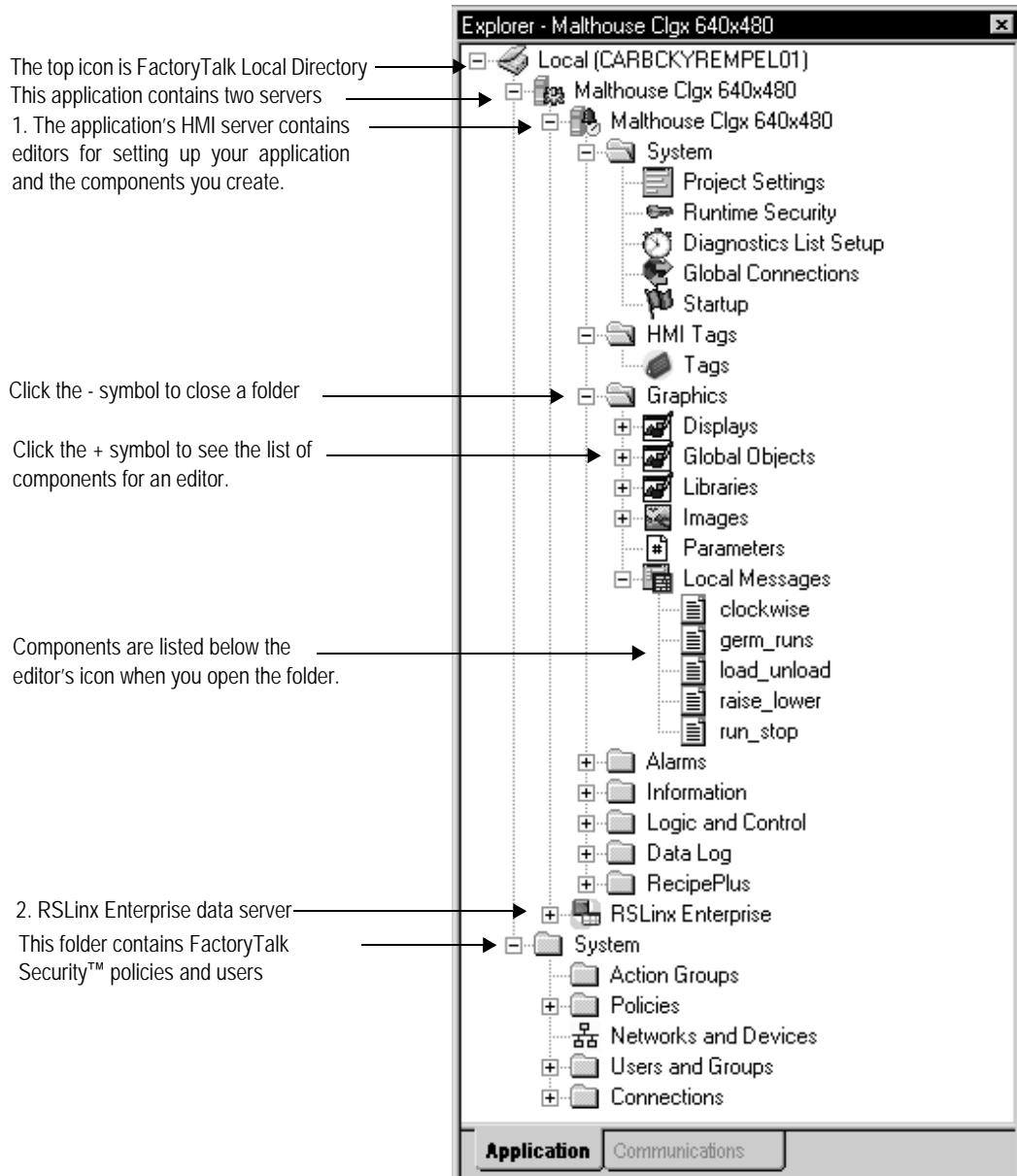
### Using the Explorer window

The Explorer window is the main tool for working with FactoryTalk View Studio. It lists the editors you use to develop your application, as well as the components, such as graphic displays, you've created.

The Explorer window also allows you to set up security for your application. You can use the window to set up users and user groups, and to assign security access to FactoryTalk® Directory to the application, and to networks and devices. For information about setting up security, see Chapter 11.

An application consists of one or more data servers and an HMI project (also known as an HMI server). The data servers provide communications for the project. The project consists of graphic displays, alarm information, user information, and other settings.

This manual generally uses the term application to refer to both application-level and project-level procedures.



You can resize the Explorer window by dragging its edges.

## **Viewing the Explorer window**

You can view the Explorer window in a number of ways:

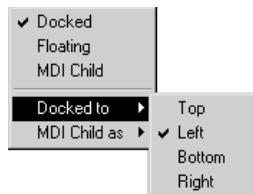
- As a docked window. When docked, the explorer is always on top of other windows that it overlaps. You can dock the explorer to any edge of the main window.
- As a floating window. When floating, the explorer is always on top of other windows that it overlaps, but you can move it to different locations in the middle of the main window.
- As a “child” window. When the explorer is in child mode, you can minimize the explorer or position other windows in front of it. This mode is particularly useful when you are working in multiple editors or working with the Help open.

To put the explorer in child mode, right-click the Explorer window title bar, and then click MDI Child.

## **Moving the Explorer window**

You can move the Explorer window in a number of ways:

- Click the title bar and drag.
- Right-click the title bar, then select a new docking location, floating, or child mode.



## **Working with editors**

When developing an application, you will use many different editors, but they have many similar features and often require similar information. Knowing how to use these features saves time.

For information on working with particular editors, see the chapters later in this guide.

## **Locating editors**

The editors are grouped in folders in the Explorer window (see the illustration on page 2-9). Each editor is displayed with an icon to the left of the editor name.



## To open a folder in the Explorer window

1. Click the + symbol to the left of the folder icon, or double-click the folder name.

## Viewing an editor's components

With some editors you enter information in a single window or a tabbed dialog box. Other editors allow you to create multiple components, such as graphic displays or message files. Each component is stored in a separate file, with its own name.

You can create components in these editors:

- Graphics (components include graphic displays, global object displays, and graphic libraries, each in their own folder)
- Parameters
- Local Messages
- Information Messages
- Data Log Models
- Macros
- RecipePlus

The Explorer window lists the components you create under the icon for the editor you used to create the component.

You can use the Images editor to copy bitmap images into your application (but not to create new images). Each image you copy is listed as a component under the editor.

## To view a list of components for an editor

1. Click the + symbol to the left of the editor icon, or double-click the editor name.

For information about working with components, see Chapter 18 in Volume 2 of the *FactoryTalk View Machine Edition User's Guide*.

## Opening editors

### To open an editor

1. Right-click the editor, and then click Open or New.

If there are no components in the editor, you can double-click the editor to open it.

## Closing editors

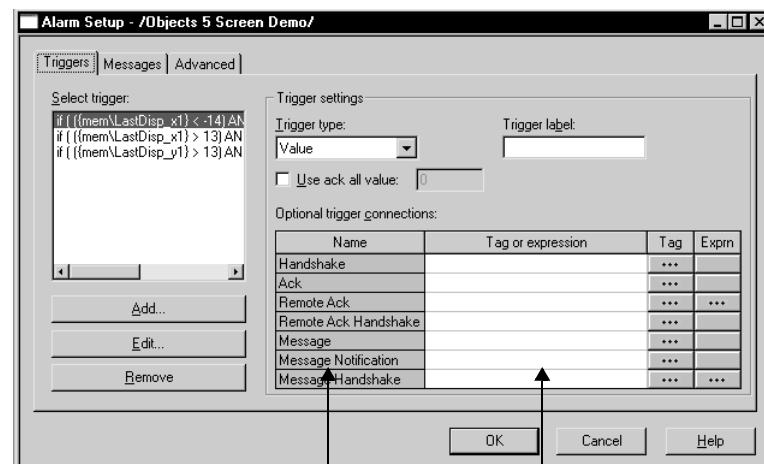
### To close an editor

1. Click the close button on the editor's title bar, or on the File menu click Close.

Some editors have an OK, Cancel, or Close button that you can click to close the editor.

## Entering information in spreadsheets

Some editors contain spreadsheets for entering information. You can enter information in the white columns. Gray columns display information; you cannot enter information in them.



This column is for display purposes only.

Enter information in this column.

These editors contain spreadsheets:

- Tags\*
- Runtime Security (formerly User Accounts)\*\*
- Global Connections
- Local Messages
- Alarm Setup
- Information Setup
- Information Messages
- Macros
- RecipePlus Editor



\* You cannot make changes in the spreadsheet section of the Tags editor. In this editor, you must use the Form section of the editor to create and edit tags.

\*\* The Runtime Security editor has two tabs. You can use the spreadsheet in the ME Runtime 3.20 and earlier tab, but not in the ME Runtime 4.00 and later tab. Use this editor to add user accounts.

In the Graphics editor, many of the Properties dialog boxes for setting up graphic objects contain spreadsheets for assigning tags or expressions to the objects.

### **To enter information in a cell in a spreadsheet**

1. Click the cell, and then type the information. If the cell already contains text, the new information is added after the current text.

In the Property Panel, double-click the cell and then type the information. If the cell already contains text, double-clicking highlights the text and typing replaces the highlighted text.

### **To move to the next cell in the row**

1. Press Tab or Enter.

The method to use depends on which editor you're working in.

### **To move to the first cell in the next row**

1. Press Enter or Down Arrow.

The method to use depends on which editor you're working in.

### **To delete a cell's contents**

1. Click the cell, and then press Delete or Backspace. Pressing Backspace removes the characters one by one.

The method to use depends on which editor you're working in.

### **To delete rows**

1. Select one or more rows, right-click the selection, and then click Delete Rows.

## **Printing**

Each editor has a Print item on its File menu.

### **To print an editor's contents**

1. Open the editor.
2. On the File menu, click Print.

3. Click OK.

## Selecting a printer

You must install a printer before you can select it. For information about installing a printer, see your Windows documentation.

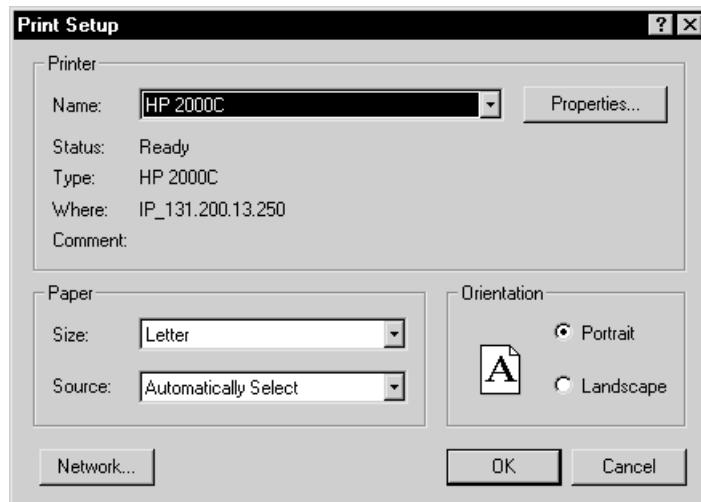


The Print Setup menu item in FactoryTalk View Studio applies to your development computer only.

To print at run time, set up a printer on the computer you'll be using to run your application. For more information about printing at run time, see page 2-15.

### To select a printer

1. On the File menu in any editor or component, click Print Setup.



2. If you don't want to use the default printer, specify another printer.
3. Choose the appropriate paper and orientation options.
4. Click OK.

## Selecting printer setup options

For detailed information about printer options, refer to your Windows documentation.

## Selecting a network printer

FactoryTalk View Studio can print to a network printer. For detailed information about setting up network printers, refer to your Windows documentation.



## Printing at run time

You can print graphic displays at run time using the display print button or using a remote display print. You can also send diagnostics and alarm messages to a printer at run time.

For information about	See
Display print buttons	page 19-30
Printing graphic displays	page 19-30
Printing displays remotely	page 8-5
Printing alarm messages	page 9-10
Printing diagnostics messages	page 10-8
Specifying the printers to use at run time for Windows 2000, Windows XP, or Windows Server 2003 R2 applications	page 15-10
The type of printer to use with a PanelView <sup>TM</sup> Plus or PanelView Plus CE terminal	page 16-1





# Planning applications

A well-designed application requires planning. To help you plan, this chapter describes:

- understanding the machines and processes you're automating.
- collecting data.
- designing an HMI tag database.
- planning graphic displays and navigating between them.
- planning languages.
- planning alarms.
- providing information for the operator.
- planning trends.
- planning recipes.
- designing a secure system.

FactoryTalk® View comes with several sample applications that can give you guidance when planning your application. For instructions on opening the sample applications, see page 2-2.

## Understanding the process

Gather information from a variety of sources so you have a complete and detailed understanding of the machines and processes you're automating.

To gather information:

- Talk to machine operators and other experts who are going to use the system. Find out what information they need to optimize machine operations and processes.
- Talk to management and management information systems staff to find out what information they need to support planning decisions.
- Break up each section of the process into its constituent parts.
- Determine what type of communications you'll be using—which network types, data servers, and devices.
- Determine which process variables you need to have access to and identify their locations in the programmable controllers.

For information about setting up communications, see Chapter 5.

## Collecting data

When planning data collection, design your system so only essential data is collected. Limiting data collection is important because collection activities require substantial processing power and generate a lot of traffic on the network.

Keep data collection requirements in mind when designing the layout of the programmable controller data tables and the HMI tag database. Ideally, tag addresses reference contiguous blocks of programmable controller data tables to reduce highway traffic and optimize system response. If you're going to use data from an OPC® server, read the server documentation to find out how to optimize traffic.

## Designing an HMI tag database

Take the time to plan your HMI tag database. A good design helps reduce the time required for maintenance and can improve programmable controller-to-FactoryTalk View response time.

For information about creating HMI tags, see Chapter 7.

If you are going to use direct referencing for all your tags, you don't need to create HMI tags in FactoryTalk View.



Direct reference tags usually update faster than HMI tags. To obtain the best system performance for your application, use direct reference tags where possible.

For information about creating direct references to data server tags, see Chapter 6.

## Collecting information

Before you begin creating the tag database, collect:

- flowcharts of your processes (or process and instrument diagrams).
- a list of programmable controller data table or register addresses your application will have access to.
- machinery systems documentation.
- alarm requirements (for more information, see page 3-5).

## Organizing tags

Before creating tags:

- Develop naming conventions for tags, choosing names that are familiar and logical to everyone. This makes troubleshooting easier.
- Group related tags.

Group tags in the way that makes the most sense for your application. For example, group all similar devices or group related areas of the plant floor.

When creating tags, place related tags into folders. For greater organization, nest folders.

## Planning graphic displays

When planning displays, determine the best way for users to navigate through your displays and develop a template to establish a common look and feel for your displays.

Also consider factors about the runtime terminal that affect how the application is displayed and used, such as:

- screen size.
- whether the operator will be using a keyboard, touch screen, mouse, or combination of navigation and data entry methods.

For information about creating graphic displays, see Chapter 19.

## Developing a hierarchy of displays

A hierarchy of displays is a series of graphic displays that provide progressively more detail as users move through them. Design your display hierarchy to meet the needs of the various users, including managers, supervisors, and operators.

Well-organized graphic displays present information clearly and consistently and guide users through the system. Before designing individual graphic displays, plan an overall display hierarchy and plan how users will navigate through the hierarchy.

For information about navigation methods and developing a hierarchy of displays, see Chapter 8.

## Creating a template to ensure consistency

It is possible to keep a consistent appearance among all the displays in an application by presenting the same pieces of information in the same place on each graphic display. To ensure uniformity, develop a display with common elements that acts as a template. Each time you develop a new display, start with a duplicate of the template.

For example, the template could contain:

- your company logo.
- a title.
- the date and time.
- navigational buttons.

Use global object displays to create template objects. You can copy the objects into your graphic displays, and update all the objects at once by updating the original object in the global object display. For information about creating global object displays, see page 25-7.

## Designing displays

When designing displays, applying good visual design principles helps users and increases their efficiency. Remember these important design principles:

### Consistency

- Be consistent with your use of symbols and color.
- Be consistent with button labels and button placement.

When you design several displays, place the same kinds of buttons in the same positions. For example, if you have a Start button in a certain position in one display, don't put a Stop button in the same position in the next display.

### Clarity

- Use symbols that are easily recognizable. For example, use the conventional ISA symbols for tanks and valves.
- Don't overload the screen with information.
- Use standard, clear terminology, and avoid abbreviations or acronyms that the user might not understand.
- Use colors with recognizable meanings. For example, in North America the colors red and green usually mean stop and start. Keep color meanings consistent by assigning red only to Stop buttons, and green only to Start buttons.

Some people are color blind to red and green so don't rely on color alone to establish meaning.

- Use high contrast color combinations, such as yellow on blue.

### Usability

- If you're designing for a touch screen, don't place important buttons where they'll be blocked by an On Top display. The user can't press a covered button. Also, ensure the button is large enough for users to touch easily.
- Ensure there is always a clear way to move between displays.
- Keep the intended user in mind and design displays so they are easy to understand and use. Ask the users to test the displays.

## Planning languages

Before setting up languages, plan:

- which languages you need to use.
- which Windows fonts support these languages.
- how different languages will affect design elements in your graphic displays, such as object size and message length.
- whether operators need to switch languages at run time, and if so, under what conditions. This will help you determine where to locate language switch buttons in your application.
- how to show operators which button to press to switch to their languages. For example, by using a text label in French, or a French flag, to alert a French operator.

For information about setting up language switching, see Chapter 12.

## Planning alarms

Before setting up alarms, plan:

- what conditions will trigger alarms.
- how operators will be notified of alarms.
- what information you want alarm messages to contain.
- how operators will respond to alarms.

For information about setting up alarms, see Chapter 9.

## Providing information for the operator

In addition to notifying the operator of alarm conditions, you can provide information and instructions about plant processes, and inform the operator about system activities.

Before setting up information notification, plan:

- what system activity the operator needs to be informed about.
- what conditions will trigger information messages.
- what information you want the messages to contain.

## Local and information messages

Use local messages to give the operator information in a specific graphic display while the display is open. Use information messages to give the operator information no matter which display is open.

For information about setting up local messages, see page 19-25. For information about setting up information messages, see Chapter 27.

### **Diagnostics messages**

Set up diagnostics messages to notify the operator of system activity such as tag reads and writes, running macros, communication problems, or problems opening displays.

For information about setting up diagnostics messages, see Chapter 10.

### **Planning trends**

When planning trends, consider how they will be used. For example, will the trend be used to:

- analyze process trends?
- monitor production efficiency?
- archive process variables to ensure compliance with government regulations?

Based on such considerations, you can determine:

- which tags need to be plotted on the same trend.
- which tags need to be logged by using a data log model.

For information about setting up data logging, see Chapter 26. For information about creating trend graphic objects, see Chapter 28.

### **Planning recipes**

When planning recipes, consider how they will be used. For example, will recipes be used to:

- compare tag values to pre-set data values?
- upload tag values to existing or new data sets?
- download values from data sets to tags?
- archive tag values to ensure compliance with government regulations?

Based on such considerations, you can determine:

- which tag sets and data sets to pair together in a recipe file.
- which RecipePlus button actions to use with the RecipePlus selector and RecipePlus table.

For information about using recipes and creating RecipePlus graphic objects, see Chapter 29.

## Designing a secure system

When deciding on your security requirements, consider whether to:

- restrict access to every graphic display to prevent accidental changes.
- restrict access to certain graphic displays.
- have everyone log on.
- use passwords, and if so, whether to have a minimum length, and whether to require that the passwords are changed periodically.
- allow only authorized users to shut down the application.
- use the DeskLock tool to prevent users from switching to another application at run time (for applications that will run on personal computers). You can use this tool to set up a customized desktop and to prevent access to Windows® functions.

Based on these considerations, you can set up security for individual users or groups of users. For example, you might want to set up groups of users such as a manager group and an operator group.

For more information about setting up security, see Chapter 11.



 4

# Working with applications

This chapter describes:

- application files.
- creating, importing, opening, and closing applications.
- renaming, copying, deleting, backing up, and restoring applications.
- specifying project settings.
- viewing application properties.
- viewing HMI server properties.

## What is an application?

An application is the software application you create in FactoryTalk® View Studio to monitor and control your plant processes. An application consists of one or more data servers and an HMI project (also known as an HMI server). The data servers provide communications for the project. The project or HMI server consists of graphic displays, alarm information, user information, and other settings.

For information about viewing the application properties, see page 4-17. For information about viewing the HMI server properties, see page 4-17.

## Application versus project

This manual generally uses the term application to refer to both application-level and project-level procedures. An exception is the section on using the Project Settings editor, which begins on page 4-11. In that section, the term project is used.

When you create an application, FactoryTalk View creates folders and files in various locations on the development computer. Some of the folders are empty until you start setting up your application. When you finish developing the application, FactoryTalk View uses the information in the various folders and files to create the runtime application.

## HMI project file

The HMI project file has the extension .med. The HMI project file is located in this directory:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\ME\HMI projects (Windows® 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\ME\HMI projects (Windows XP or Windows Server 2003 R2)

## Runtime application file

The runtime application consists of a file with the extension .mer. For information about creating the runtime application, see Chapter 14.

## Component files

Component files are located in folders in the HMI project folder, as described in the following table. For example, graphic display component files (with the extension .gfx) are stored in the Gfx folder. You might need to navigate to the location of these files, for example if you are adding a component from one application into another application.

This folder	Contains	File extensions
Application name	HMI project file and the folders described below	.med
DLG	Data log models (component files)	.mdf
Gfx	Graphic display component files	.gfx
Global Objects	Global object display component files	.ggfx
Images	Image files	.bmp or .jpg
Information	Information message component files	.ifm
Local	Local message component files	.loc
Macros	Macro component files	.mcr
PAR	Parameter component files	.par
RecipePlus	RecipePlus component files	.rpp

## External folders

The following table lists some FactoryTalk View folders that are external to the HMI project folder, and the types of files they contain.

The folders are located in this directory:

C:\Documents and Settings\All Users\Documents\RSView Enterprise (Windows 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise (Windows XP or Windows Server 2003 R2)

This folder	Contains	File extensions
Images	Image files for use on graphic objects	.bmp
ME\Libraries	Graphic library component files and image files used in the library displays	.gfx .bmp
ME\Logs\Runtime <i>Application Name</i> \ Dlglog\Data Log Model <i>Name</i>	Data log files for applications that have run on the development computer	.log .tag
ME\Logs\Runtime <i>Application Name</i> \ M_Alarms	Alarm log file for applications that have run on the development computer	.alm
ME\Runtime	Runtime application files	.mer
ME\HMI projects	Sample application folders, including all files and folders needed for the sample	
ME\Faceplates	Graphic displays and global object displays for Logix5000 faceplates	.gfx .ggfx .bmp

If desired, you can specify a different directory in which to store graphic library component files. For more information, see page 19-19.

## Default log file locations for PanelView Plus or PanelView Plus CE applications

On the PanelView™ Plus or PanelView™ Plus CE terminal, alarm and data log folders are located by default in this directory:

\Storage Card\Rockwell Software\RSViewME\Logs\

## Data log file locations

You can also store data log files in a different location on the runtime computer, on a networked computer, or on a PC card. For more information, see Chapter 26.

## Naming files

File names, including the path, can be up to 200 characters long (file names can contain spaces). For example, the following path and file name contains 114 characters:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\ME\HMI  
projects\Malthouse Clgx 640x480\Gfx\Steeping gfx

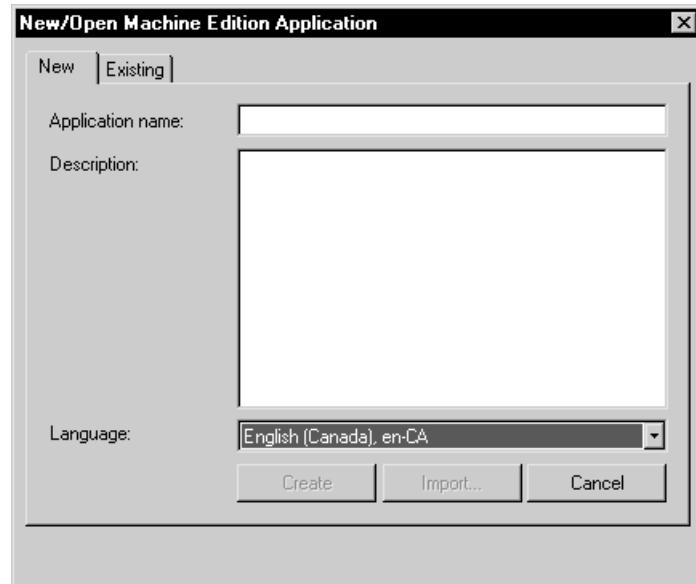
## Creating, importing, opening, and closing applications

### Creating applications

You can create a new application when you start FactoryTalk View Studio, or once FactoryTalk View Studio is already open.

### To create a new application when you start FactoryTalk View Studio

1. Start FactoryTalk View Studio.



2. In the New tab, in the Application name box, type a name for your application, up to 32 characters long.
3. If desired, type a description of the application. If you don't type a description now, you can add one later, as described on page 4-17.
4. Specify a language for the application. For information about using different languages, see Chapter 12.
5. Click Create.

FactoryTalk View Studio creates the application's folders and files, and then displays the new application in the Explorer window.

The application is created in the ME\HMI projects directory, in a folder with the same name as the application name.

This is the path to the ME\HMI projects directory:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\ME\HMI projects (Windows 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\ME\HMI projects (Windows XP or Windows Server 2003 R2)

 Once you create the application, specify the project settings. These settings determine the general appearance of your runtime application, and affect how your graphic displays look. For more information, see page 4-16.

### To create a new application when FactoryTalk View Studio is already open



New Application

1. On the File menu, click New Application, or click the New Application tool.

If an application is already open, FactoryTalk View Studio asks you whether to close the application that is currently open. Click Yes.

2. Follow steps 2 through 5 in the previous procedure.

### Importing applications

You can import PanelBuilder™, PanelBuilder32, PanelBuilder 1400e, and FactoryTalk View Machine Edition applications into FactoryTalk View Studio.

PanelBuilder and PanelBuilder32 files can be in either development format (\*.pba) or runtime terminal format (\*.pva). PanelBuilder 1400e files have the file extension \*.pvc.

When you import older application files, FactoryTalk View creates a copy of the application for the current version of FactoryTalk View. The older version is not altered.

You can use current version application files to create previous version runtime files. For more information, see page 4-9.

If the application you are importing contains any direct driver nodes, the import wizard will create one RSLinx® topic for each direct driver node it finds. All topics are then converted into device shortcuts, to run with RSLinx® Enterprise™. You must have both RSLinx® Classic™ and RSLinx Enterprise installed to make this two-step conversion.

To create RSLinx topics after import, follow these steps in RSLinx before importing the application.

### **To validate direct driver nodes in RSLinx**

1. Create the appropriate RSLinx Classic drivers if not already created (for example, AB\_ETH-1).  
For more information about creating RSLinx Classic drivers, see the RSLinx Classic Help.
2. If it is an Ethernet® driver (AB\_ETH-1), add the IP address of each direct driver node to the Ethernet driver setup.
3. Start RSWho and select each device associated with each direct driver node (for example, 131.200.13.128).

If these steps are not followed in this order, the topics generated by the import code will be invalid and will need to be fixed using the Topic Editor in RSLinx.

### **To import an application**

1. Start FactoryTalk View Studio
2. In the New tab, in the Application name box, type a name for the imported application, up to 32 characters long.
3. If desired, type a description of the application. If you don't type a description now, you can add one later, as described on page 4-17.
4. Specify a language for the application. For information about using multiple languages, see Chapter 12.
5. Click Import. The Machine Edition Import Wizard opens.
6. Follow the directions in the Machine Edition Import Wizard.

For more information about importing PanelBuilder 1400e applications, see Appendix A. For more information about importing PanelBuilder and PanelBuilder32 applications, see Appendix B.

## Opening applications

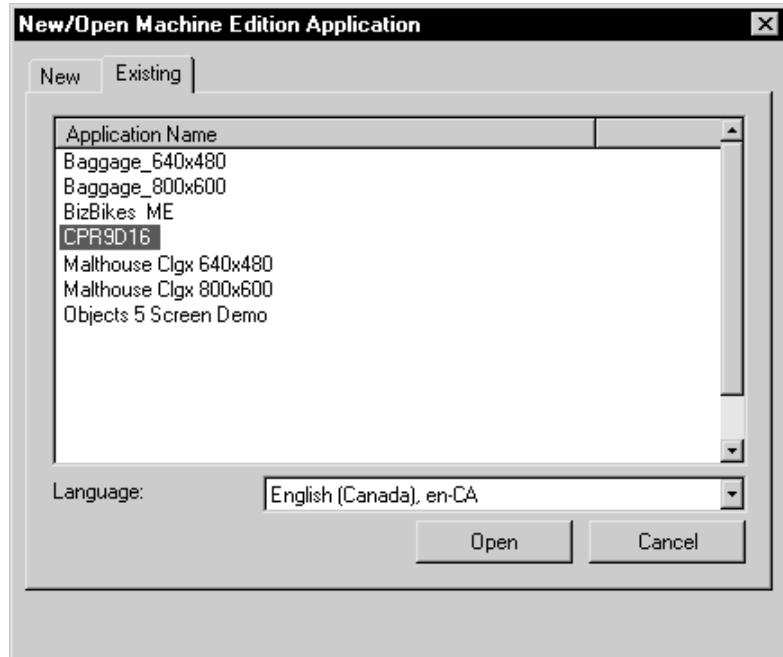
You can open an application when you start FactoryTalk View Studio, or once FactoryTalk View Studio is already open.

Older RSView Machine Edition application files are converted to the current version of FactoryTalk View when you open them. However, you can still use the current version of FactoryTalk View Studio to create older version runtime files. For information about the features supported in different versions of FactoryTalk View, see Appendix G.

For information about opening a sample application, see page 2-2.

### To open an existing application when you start FactoryTalk View Studio

1. Start FactoryTalk View Studio.



2. Click the Existing tab, and then click the application to open.

3. Specify a language for the application, and then click Open. For information about using different languages, see Chapter 12.
4. If this is the first time you are opening an application that was created in RSView Machine Edition 4.00, you are prompted to select a default language. Select a language, and then click OK. For information about the default language, see page 12-2.

FactoryTalk View Studio displays the application in the Explorer window.

### **To open an existing application when FactoryTalk View Studio is already open**



Open Application

1. On the File menu, click Open Application, or click the Open Application tool.

If an application is already open, FactoryTalk View Studio asks you whether to close the application that is currently open. Click Yes.

2. Follow steps 2 through 4 in the previous procedure.

### **To open an application you used recently**

1. On the File menu, select Recent Applications, and then click the name of the application to open.
2. If an application is already open, FactoryTalk View Studio asks you whether to close the application that is currently open. Click Yes.

The application opens. For applications with multiple languages, the language you used most recently opens. For information about using different languages, see Chapter 12.

### **Problems opening applications**

If you have trouble opening an application in FactoryTalk View Studio, the problem could be a Microsoft® Windows security setting. You must have read and write access to the folder that contains the HMI projects. For example, if you belong to the Windows user group called Power Users, and the group does not have write access to the HMI projects folder, you won't be able to open applications in FactoryTalk View.

The default location for the HMI projects folder is:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\ME  
(Windows 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\ME  
(Windows XP or Windows Server 2003 R2)

For more information about security, see Chapter 11.

### To set up write access for Power Users

1. Right-click the HMI projects folder, and then click Properties.
2. In the Security tab of the Properties dialog box, select Power Users from the list of groups and user names.
3. In the Permissions box below the list, select Full Control, and then select Allow.

### Opening multiple applications

To open two applications at the same time, for example to copy images and graphic objects between applications, start two instances of FactoryTalk View Studio.



To view a graphic display in different languages at the same time, open the application multiple times, selecting a different language each time you open it.

### Opening and editing applications from earlier versions of RSView

You can open and edit applications from RSView 3.00, 3.10, 3.20, and 4.00. When you are finished editing, you can create runtime application files for RSView ME Station version 3.00, 3.10, 3.20, or 4.00. This allows you to continue running your applications on existing terminals, without upgrading all your software.

The application is converted to FactoryTalk View 5.00 when you import it, but when you create the runtime application file you can save it back to the earlier version.

You can even use FactoryTalk View 5.00 global objects, and create multiple language versions of your older applications. When you create the runtime application file, it uses the application's current language. You can create as many different runtime application files, in as many different languages, as you want. However, the language switch feature is not available at run time for version 3.00, 3.10, and 3.20 applications. Global objects are converted to regular objects in the runtime application file for versions 3.00, 3.10, and 3.20.

For information about importing applications, see page 4-5. For information about the features supported in different versions of FactoryTalk View and RSView, see Appendix G. For information about using multiple languages, see Chapter 12. For information about using global objects, see Chapter 19.

## Closing applications

### To close an application

1. On the File menu, click Close Application.

If there are any unsaved changes in open editors, FactoryTalk View Studio asks you whether to save the changes before closing the application.

### To close an application and exit FactoryTalk View Studio at the same time



Close button

1. Click the Close button at the right end of the FactoryTalk View Studio title bar.

If there are any unsaved changes in open editors, FactoryTalk View Studio asks you whether to save the changes before closing.

## Renaming, copying, deleting, backing up, and restoring applications

Use the Application Manager tool to:

- rename applications.
- copy applications.
- delete applications.
- create a compressed backup copy of an application, with the file extension .apa.
- restore an application from the backup file, with the file extension .apa.
- restore a legacy project backup file (with the file extension .mea) from an earlier release of FactoryTalk View Studio, and convert it to version 5.0.
- restore a runtime application file, with the file extension .mer, to a development application, with the file extension .med. For more information, see page 14-6.

### To start the Application Manager tool, do one of the following

- In FactoryTalk View Studio, on the Tools menu, click Application Manager.
- On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk View, Tools, and then click Application Manager.

For details about using the tool, see the tool's Help.

## About project settings

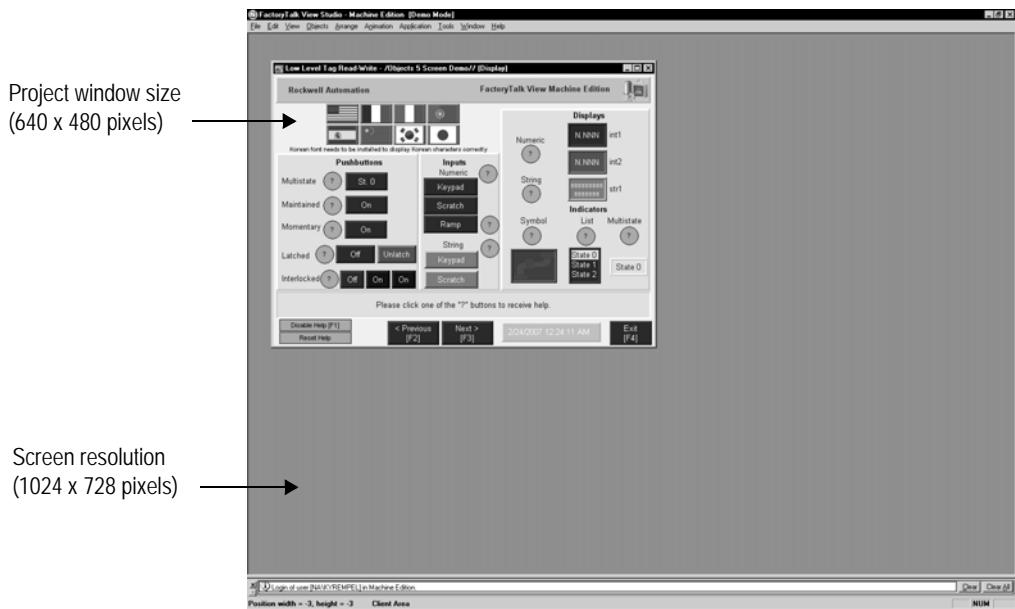
Project settings determine the general appearance of your runtime application, such as the size of the project window and whether a title bar appears in the project window.

Since project settings affect how your graphic displays look, specify the settings before creating graphic displays.

## Project window size and runtime screen resolution

This section describes the difference between project window size and runtime screen resolution, and describes how these two settings can affect the behavior of your application at run time.

The project window size is the amount of room your application occupies on the runtime terminal or computer screen, in pixels. The runtime screen resolution is the actual size of the runtime terminal or computer screen, in pixels. Therefore, the project window size must be less than or equal to the runtime screen resolution.



Usually, an application is designed to completely fill the runtime terminal or computer screen. If you want your application to do this, set the project window size to the resolution of the screen in pixels. For information on setting the project window size, see page 4-16.

## Screen resolution for PanelView Plus and PanelView Plus CE terminals

If you are going to run your application on a PanelView Plus or PanelView Plus CE terminal, select one of these project window sizes:

For this terminal or monitor	Select this project window size
PanelView Plus 400 or 600	320x240
PanelView Plus 700 or 1000	640x480
PanelView Plus CE 700H or 1000H	
PanelView Plus 1250	800x600
PanelView Plus CE 1250H	
PanelView Plus CE 1200P, 1200M, or 1200W	
PanelView Plus 1500	1024x768
PanelView Plus CE 1500H	
PanelView Plus CE 1500P, 1500M, or 1500W	
PanelView Plus CE 1700M	1280x1024
PanelView Plus CE 2000M	1600x1200 (Custom size)

If you are creating an application for a PanelView Plus 400 or 600 terminal:

- Resize the default message displays so that the entire message is visible.
- Reposition the [ALARM] display from the default position (0, 0), as this position is below the viewing area of the terminals.

## Changing the project window size after creating graphic displays

The project window size is used for all graphic displays of the Replace type. If you change the window size after you have already created graphic displays, you have the option of scaling graphic displays.

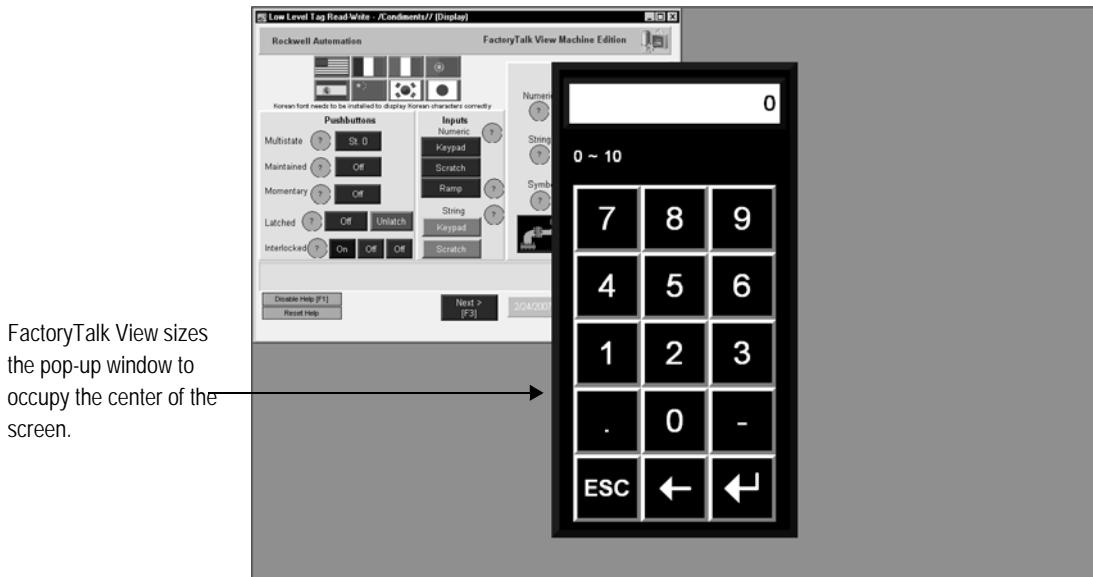
If you choose to scale graphic displays, all Replace and On Top displays are resized and the objects in them are scaled to fit the new size. You can also specify whether to scale the font size and border size of the graphic objects, and the size of graphic images in the Images folder.

If you choose not to scale displays when you change the project window size, Replace displays are resized, but any objects in the displays remain the same size and in the same position as before. On Top displays are not resized.

For more information about Replace and On Top displays, see page 19-12. For more information about scaling displays, see Help.

### How the runtime screen resolution affects the pop-up windows

FactoryTalk View comes with a pop-up Login window, as well as numeric and string pop-up windows for runtime data entry, and a Change Password window for changing user passwords at run time.



### How the project window size affects the default message displays

Your application comes with preconfigured, default graphic displays for diagnostics, alarm, and information messages. If the project window size is smaller than the default displays, the edges of the displays are cut off at run time.

These are the sizes of the default displays:

Graphic Display	Width in pixels	Height in pixels
[DIAGNOSTICS]	640	160
[ALARM]	640	210
[INFORMATION]	640	80

If your project window size is smaller than the default message display sizes, you can resize the default displays, or use your own displays instead.

### Title bar

If you select this option, a title bar is used for all Replace graphic displays. For information about graphic display types, see page 19-12.

If desired, you can also use a Control box, Minimize button, and Close button on the title bar. The Close button appears only if you select the Control box in the Project Settings editor.



If you plan to set up security for your application and want to prevent unauthorized users from stopping the application, do not use a title bar.

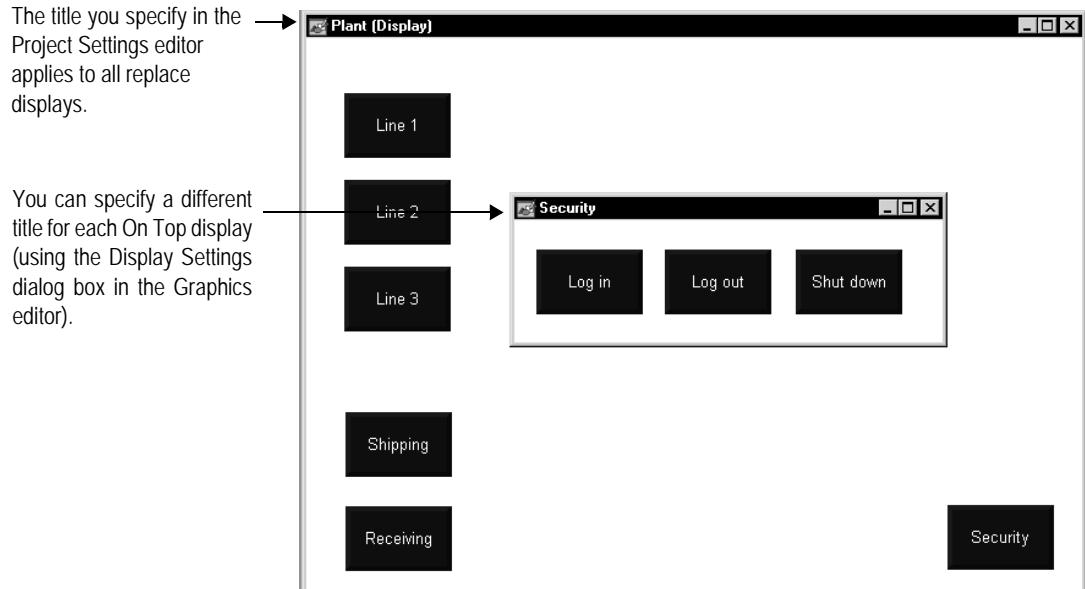
For information about setting up security, see Chapter 11.

### Changing the Title bar or Border setting

If you change the Title bar or Border setting, you have the option of scaling graphic displays to accommodate the new display size. For details, see Help.

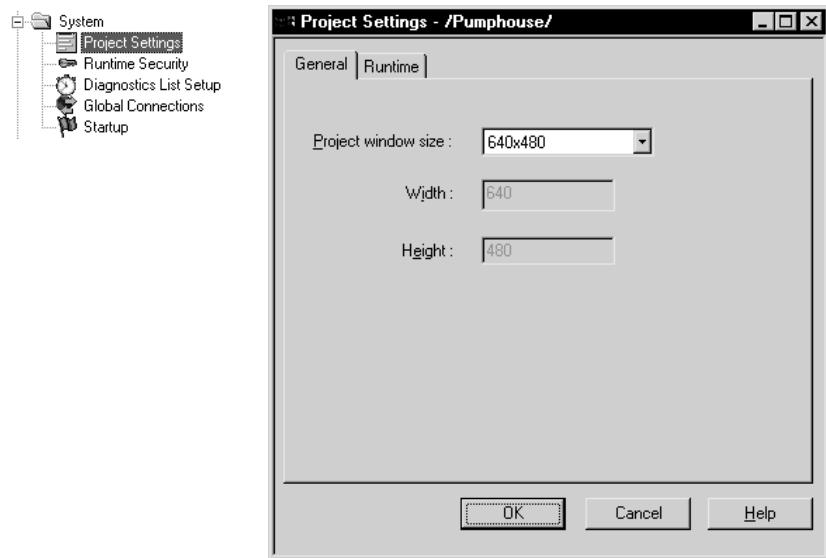
## Title bar for On Top displays

For On Top displays, you can specify a different name to use in the title bar. Use the Display Settings dialog box to specify the name (for details, see Help).



## Specifying project settings

To specify project settings, use the Project Settings editor. For details about selecting options in the editor, see Help.



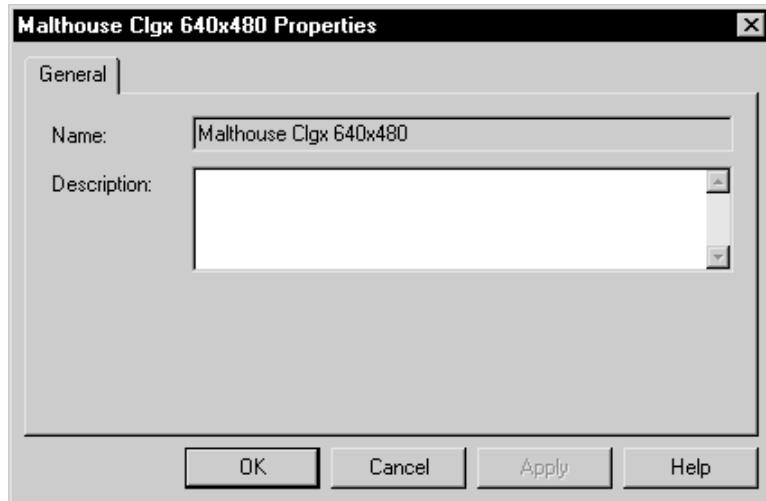
## Viewing application properties

You can view this information about your application:

- application name
- application description, if any

### To view application properties

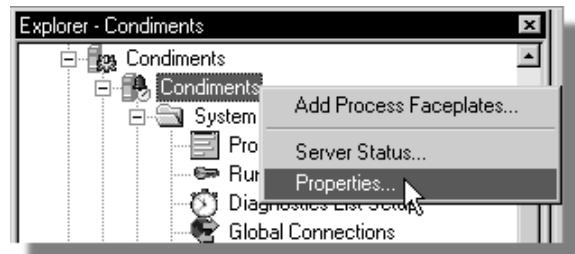
1. On the Application menu, click Application Properties.



## Viewing HMI server properties

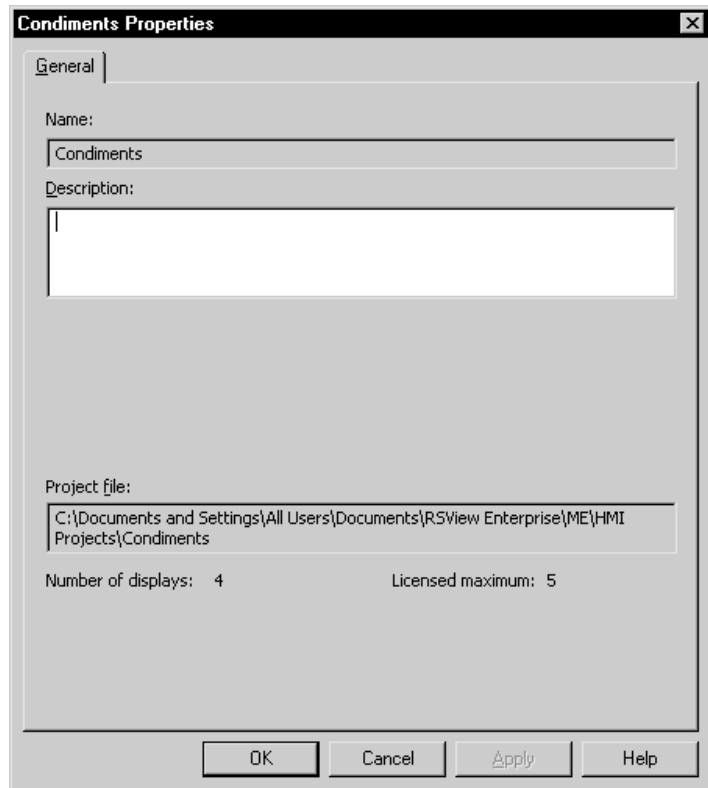
You can use the Explorer window to view this information about your project or HMI server:

- application name
- application description, if any
- location of the project file
- number of graphic displays in the application
- maximum number of displays you're licensed to use



### To view HMI server properties

1. In the Explorer window, right-click the HMI server name, and then click Properties.



# 5

# Setting up communications

This chapter describes:

- OPC® communications.
- what data servers are.
- creating data servers in your application.
- updating data server caches.

## About OPC communications

OPC® is OLE for Process Control, used to connect FactoryTalk® View to communication devices via vendor-specific OPC servers. FactoryTalk View supports the OPC Data Access (DA) 2.0 specification.

OPC servers provide a way for FactoryTalk View to retrieve tag values from:

- Allen-Bradley® controller devices, using RSLinx® Classic™ or RSLinx® Enterprise™ as an OPC server.
- third-party controller devices, such as Siemens or Modicon®, using third-party OPC servers such as KEPServerEnterprise™.



ControlNet Scheduled, DeviceNet I/O and Remote I/O are supported on PanelView™ Plus and Windows CE terminals. They are not supported on personal computers.

## Summary of steps

To set up communications, follow these steps:

1. Gather information about your network, and the devices that are connected to it. You will need this information to set up RSLinx Enterprise, RSLinx Classic, or another OPC server.
2. If you want to use RSLinx Enterprise for communications, install it on the development computer.

For information about installing RSLinx Enterprise, see the *FactoryTalk View Machine Edition Installation Guide*.

3. If you want to use RSLinx Classic as the OPC server that enables communications on your network, set up RSLinx Classic on the development computer. RSLinx Classic is included on the FactoryTalk View Machine Edition CD-ROM. For details, see *Getting Results with RSLinx*, or see Help for RSLinx.

RSLinx Classic is available for personal computers only.

4. If you want to use an OPC server other than RSLinx Classic for communications on your network, set up the OPC server on the development computer. For details, see the documentation supplied with your OPC server.
5. In FactoryTalk View Studio, create a data server that points to the OPC server you set up in step 3 or step 4.

For details, see page 5-3.

If you are using RSLinx Enterprise for communications, you don't need to create a data server because it is created automatically when you create your application.

6. If you want to use HMI tags in your application, open the Tags editor, and then create device tags that point to addresses in devices. For details about creating HMI tags, see Chapter 7.

If you do not want to use HMI tags, but instead want to reference addresses in devices directly, skip this step.

7. In objects in graphic displays, or anywhere else you want access to values in tags, type the tag name using the syntax for your OPC server, or use the Tag Browser to select tags. For details about using tags, see Chapter 6.
8. Create the runtime application file. For details, see Chapter 14.

All the necessary tag information is compiled with the runtime application file.

9. If you are using an OPC server other than RSLinx Enterprise or RSLinx Classic, install the OPC server software on the runtime computer.

For applications that will run on a personal computer, the OPC server can be on a different (remote) computer than the runtime computer, but the remote computer must be on the same network as the runtime computer.

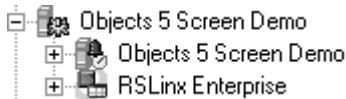
## About data servers

A data server provides a route to the physical devices on your network, so you can see the values in those devices, for example:

- programmable controller values.
- OPC tags, and their value or status information.
- named variables in a Logix5000 processor.

## Creating data servers

An RSLinx Enterprise data server is created automatically with a new application. It is located under the application's HMI server in the Explorer window, just above the FactoryTalk System folder.



You can create additional OPC data servers for your application. For example, you might want to use an RSLinx Enterprise data server and a KEPServerEnterprise data server.



If you use RSLinx Enterprise, you cannot use Logix5000 with an Ethernet connection to connect to other Allen-Bradley devices or bridge to DH+ or Remote I/O networks. If you need to do this, use RSLinx Classic.

## Setting up RSLinx Enterprise data servers

Use an RSLinx Enterprise data server when you want to use RSLinx Enterprise communications.

You can use the Communication Setup editor to add drivers, add devices, set up driver and device properties, and set up device shortcuts.

### To set up communications in RSLinx Enterprise

1. In the Explorer window, open the RSLinx Enterprise data server.



2. Double-click the Communication Setup editor.
3. In a new application, the RSLinx Enterprise Configuration Wizard opens. Follow the instructions to create a new configuration or use an existing device configuration. Once you make a choice and click Finish, the Communication Setup editor opens.

In an existing application, double-clicking Communication Setup opens the Communication Setup editor, with its two tabs. The Design (Local) tab is for establishing the location of the tags/addresses for editing. The tags/addresses can be online with a controller or other data server, or an offline controller file. This will enable the Tag Browser to find the tags/addresses.

The Runtime (Target) tab is for identifying the connection from the run-time computer or terminal to the controller or other data server. If the paths are the same, use the Copy button to copy the Design configuration to the Runtime tab.

For more information, see Help for RSLinx Enterprise.

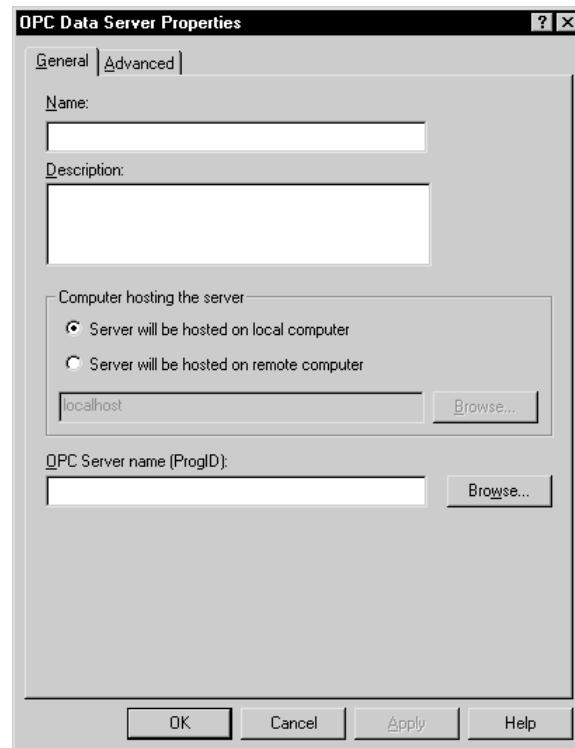
## Setting up an OPC data server

Use an OPC data server if you want to use RSLinx Classic instead of RSLinx Enterprise, or if you want to use a third party OPC data server.

You can set up a cache for tags on the data server. A cache allows you to view tag names when you are not connected to the data server.

### To create an OPC data server

1. In the Explorer window, right-click the application, select Add New Server, and then click OPC Data Server.



For details about selecting options in the OPC Data Server Properties dialog box, see Help.

For information about editing and removing data servers, see Help.

### **Updating data server caches**

If tags are added, modified, or deleted on the data server, you must update (synchronize) the cache manually.

For information about updating the data server cache periodically after you create it, see Help.





# Working with tags

This chapter provides general information about working with all kinds of tags. This chapter describes:

- types of tags.
- when to use data server tags.
- steps for using data server tags.
- when to use HMI tags.
- steps for using HMI tags.
- browsing for tags and offline tags.
- using the Tag Browser.
- using tags and expressions in your application.
- logging tag values.
- using macros to assign values to tags.

For information about creating HMI tags, see Chapter 7.

## Types of tags

A tag is a logical name for a variable in a device or in local memory (RAM). For example, a tag can represent a process variable in a programmable controller.

## Data server tags

FactoryTalk® View conforms to the OPC® Data Access (DA) 2.0 specification for information exchange among automation or control applications, field systems or devices, and business or office applications.

In FactoryTalk View, data servers such as RSLinx® Enterprise™, RSLinx® Classic™, and KEPServerEnterprise™ provide access to OPC-DA-compliant tags. In the FactoryTalk View documentation, the tags you use through a data server are called data server tags. For more information about data servers, see Chapter 5.

Data server tags include tags found in Logix5000 processors and tags from other OPC-compliant devices. You use data server tags by providing a direct reference to the tag's location, wherever you want your application to use the data.

## HMI tags

In addition to direct referencing tags from data servers, FactoryTalk View allows you to create tags with additional properties such as minimum and maximum values, scale, and offset. These tags can reference values at an external data source, or store values in the runtime computer's memory. Tags you create in FactoryTalk View are called HMI tags.

### The data source

The FactoryTalk View documentation uses the term data source as a generic term that includes all possible sources of tag data, for both data server tags and HMI tags. The data source can be memory or a device such as a programmable controller or an OPC server. FactoryTalk View writes values to and reads values from the data source. The data source is set up to exchange information (in the form of numeric or string values) between the FactoryTalk View value table and the machine that your application is controlling.

### Basic steps for using tags

To use tags, follow these basic steps:

1. If the tag does not already exist, create the tag.

To use a data server tag, you can use an existing tag in the processor (for example, a programmable controller), or you can create a new one in the processor or OPC server. For example, in a Logix5000 processor, you could create the tag using your RSLogix™ 5000 programming software.

To use an HMI tag, you must first create it in the Tags editor. For details, see Chapter 7.

2. Browse for, or type, the name of the tag anywhere you want to connect an object to data at run time.

For example, to make a push button change a tag value when it is pressed, connect the push button to a tag by typing the tag name in the Connections tab of the push button's Properties dialog box.

### Browsing for tags

If you don't know the names of tags, you can browse for them. You can browse while online and connected to a device, or you can browse for tags from an offline file, for example a Logix5000 program file.

To browse for tags, use the Tag Browser. For details, see page 6-5.

## Using tag names that don't exist

You can type the name of a tag that doesn't yet exist. If you do this, ensure that you spell every instance of the tag name consistently, and that when you do create the tag, you spell it the same way you did when you referred to it.

## When to use data server tags

For some of the things you might need to do in your application, you might not need to create HMI tags. Use direct references to tags located in devices instead, for example, tags located in an OPC server's database.

## Eliminating duplication

Using data server tags allows you to add, modify, or delete tags in a device without having to duplicate the changes in the FactoryTalk View HMI tag database.

## Using complex data

Some devices, for example Logix5000 processors, support complex data types such as arrays and structures. Your controller might have structures that contain hundreds of member elements.

Referencing tag values directly eliminates the need for creating an HMI tag for each member.

## Steps for using data server tags

To use data server tags, complete three basic steps:

1. Create the tag in the OPC server or processor, or use an existing tag in the processor.

The server must understand the syntax of the device it is connected to. You need only create a tag if you want to browse the server.

Examples of OPC servers include RSLinx Enterprise, RSLinx Classic, and KEPServerEnterprise. For information about creating tags in OPC servers that have their own tag database, see the documentation provided with the server.

For information about creating tags in a Logix5000 processor, see the documentation for your device programming software.

2. Create a data server in FactoryTalk View, if your application doesn't already have one.

For more information, see Chapter 5.

3. In graphic displays, data log models, or anywhere else you want to use a tag's value, create a direct reference to the tag. At run time, the tag's value is passed to the graphic display or data log model.

## When to use HMI tags

Use HMI tags to manipulate tag data and to store tag values in the runtime computer's memory.

## Scaling, offsetting, or providing a range for data

Use HMI tags if the data server you are using does not provide for:

- scaling or offsetting a value.
- setting minimum or maximum limits on a value (that is, providing a range of values).

## Scale and offset

The scale and offset modify the "raw data" that comes from and goes to the programmable controller before the data is saved in the computer's memory (called the value table). The scale and offset also modify the value specified in FactoryTalk View before it is written to the programmable controller.

The scale is a multiplication factor—the value from the programmable controller is multiplied by the scale.

The offset is a fixed value—after the value from the programmable controller is multiplied by the scale, the offset amount is added.

## Minimum and maximum

HMI tags allow you to set a minimum and maximum value that can be written to the programmable controller or server.

These values do not affect what is read from the programmable controller or server. For example, if you specify a minimum of 0 and a maximum of 100, FactoryTalk View would be able to read a value of 200 from a programmable controller and store it in the value table, but would not be able to write this value to the programmable controller.

For more information about minimum and maximum, and scale and offset, see Help.

## Storing values in FactoryTalk View memory

A memory tag can be used to store values without the need for an attached or accessible device.

For example, you might need to store a value in memory:

- to store the result of a calculation.
- temporarily, for example, a counter or index.

- to maintain information about the system's current state, for example which graphic display was last displayed.

For information about creating HMI memory tags, see Chapter 7.

## Steps for using HMI tags

To use HMI tags, complete two basic steps:

1. In the Tags editor in FactoryTalk View, create a tag and map the tag name to an OPC tag or device address.

For more information, see Chapter 7.
2. In graphic displays, alarm triggers, or anywhere else you want to use a tag's value, assign the tag. At run time, the tag's value is passed to the graphic display or alarm system.

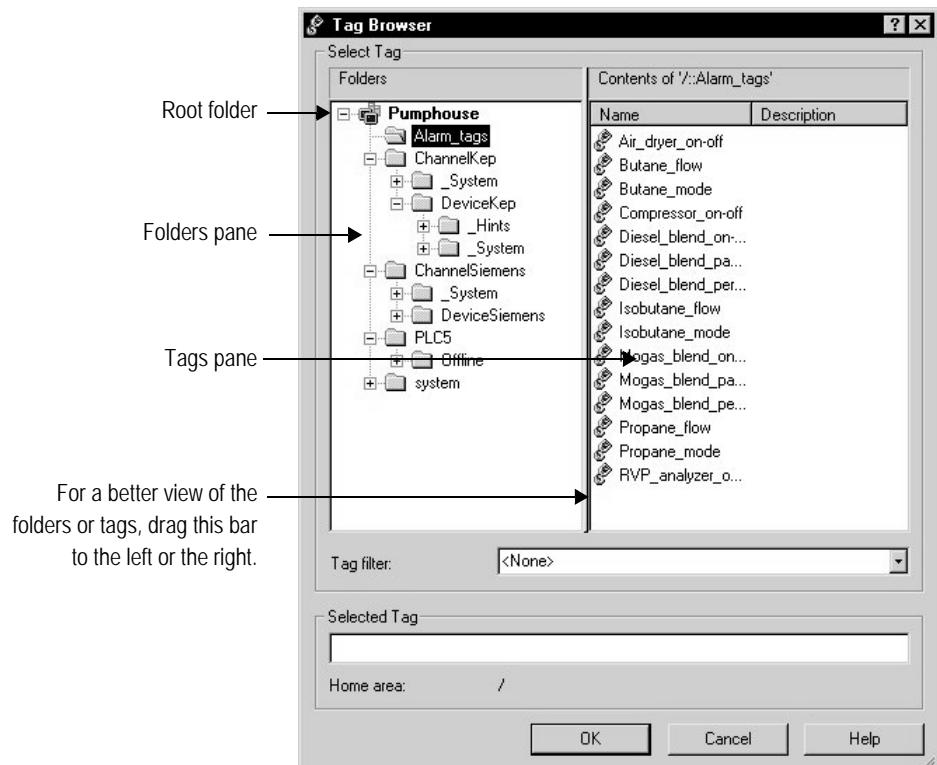
## Browsing for tags

You don't have to remember the path and name of a tag to use it in your application. Whenever you need to use a tag, you can open the Tag Browser and select the tag to use.

### To open the Tag Browser

- In most editors, click the Browse button in the Tags column.
- In the Expression editor, position the cursor where you want to insert a tag name, and then click Tags.
- In the Parameters editor, double-click where you want to insert a tag name.

## Using the Tag Browser



In the Tag Browser, you can:

- select a single tag or multiple tags. The ability to select multiple tags is available in the Data Log Models editor only. In the illustration above, only a single tag can be selected.
- select data server tags and HMI tags.
- select a folder of tags without selecting a specific tag in the folder. This is useful for applications that use process faceplates. It is also useful when you use global object parameters. Folders of tags are also known as backing tags or structured tags. For information about global object parameters and process faceplates, see Chapter 25.

You cannot select backing tags when you open the Tag Browser from the Data Log Models editor.

- create and edit HMI tags.

- create new folders for HMI tags.
- import tags from a PLC or SLC™ database.

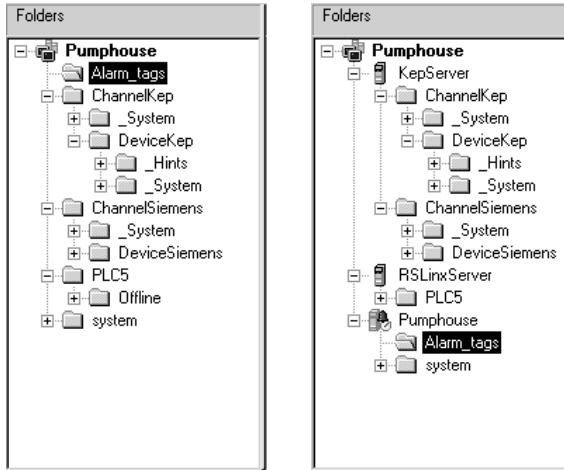
For more information about using the Tag Browser, see Help.

## Showing server names

By default, the folder pane in the Tag Browser shows folders, but not the servers they belong to. You can show the names of servers that have been set up in the application.

### To display server names

1. Right-click a blank area of the folders pane and then click Show Server Names.



The folders list with server names hidden.

The folders list with server names shown.

In the example on the left, folders are listed alphabetically by name.

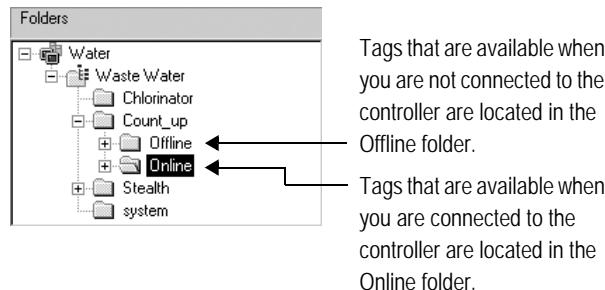
In the example on the right, there are two data servers, named KepServer and RSLinxServer. Folders of data server tags are listed under the data server to which they belong. Below the data servers, the application is listed as an HMI server, with folders of HMI tags under the application name.

## Browsing for off-line tags

For each RSLinx Enterprise device shortcut, or each RSLinx Classic OPC topic in your application, an Offline and an Online folder is displayed in the Tag Browser.

In RSLinx Enterprise, you can use the Offline folder to browse tags in an RSLogix 5000 RSLogix.acd file. For details about making the .acd file available to browse offline, see RSLinx Enterprise Help.

In RSLinx Classic, you can use the Offline folder to browse tags in a PLC program stored on disk. If the OPC topic in RSLinx Classic has access to symbols, you can browse for them in the Offline folder. For details about adding symbols to the OPC topic in RSLinx Classic, see RSLinx Classic Help.



## Using tags and expressions in your application

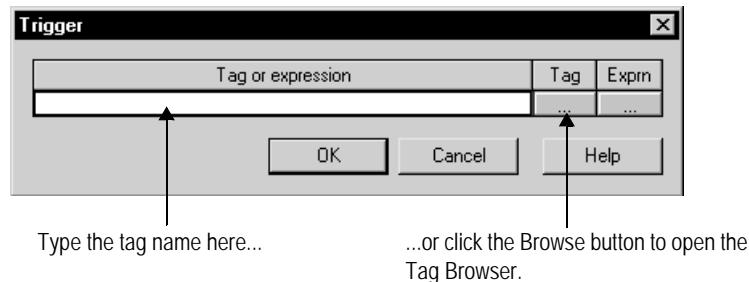
Once you've created tags (either in data servers or in FactoryTalk View), assign them to connections in your application to allow the application to interact with the data source and perform actions based on the tag values.

You can assign tags in these editors:

- Expressions
- Global Connections
- Graphic Displays
- Global Object Displays
- Graphic Libraries
- Parameters
- Alarm Setup
- Information Setup
- Macros
- Data Log Models
- RecipePlus Editor

## Assigning tags

This example shows the Trigger dialog box that opens when you click the Add button in the Triggers tab of the Alarm Setup editor:



### To assign a tag, do one of the following

- In the “Tag or expression” column, type the tag name.
- In the Tag column, click the Browse button and then select a tag from the Tag Browser.



For information about using the Tag Browser, see Help.

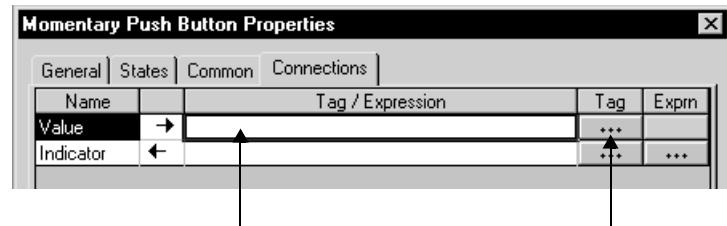
## Assigning tags to graphic objects

In the Graphics editor, you can assign tags to many of the graphic objects you create.

You can use up to 1,000 tags per graphic display. This limit includes the tags contained in embedded variables and expressions. For example, if an expression references the same tag twice, this counts as two tag references.

Use one of these methods to assign tags to a graphic object:

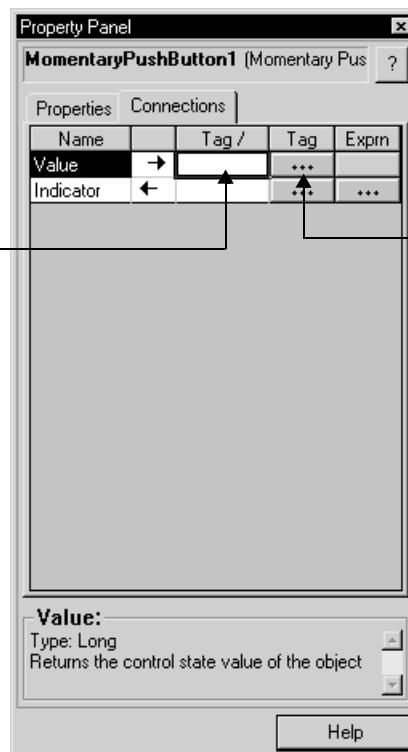
- Double-click the object to open the object's Properties dialog box, and then click the Connections tab.



Type the tag name here...

...or click the Browse button to open the Tag Browser.

- Select the object, and then assign tags in the Connections tab of the Property Panel.



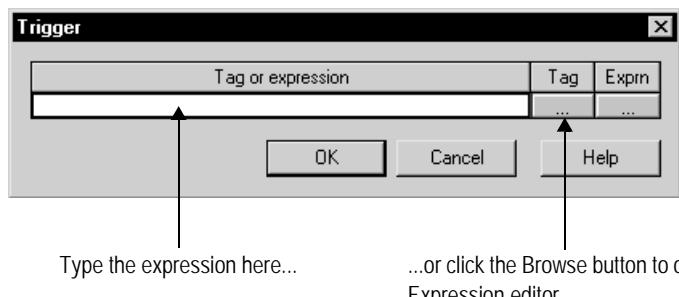
Type the tag name here...

...or click the Browse button to open the Tag Browser.

## Using expressions to manipulate tag values

Many of the connections to which you can assign a tag also permit the use of expressions to perform logical or mathematical calculations on tag values. If you assign an expression, FactoryTalk View monitors the expression value rather than the original tag value. For example, your machine might send values to the data source in the form of temperature in degrees Celsius. You could use an expression to convert the value to degrees Fahrenheit, and then monitor the expression result rather than the original tag value.

If you can assign an expression, a Browse button is present in the Exprn column in the editor or Connections tab.



### To assign an expression, do one of the following

- In the Exprn column, click the Browse button, and then create an expression in the Expression editor. Use this method if you want to check the expression syntax.
- In the “Tag or expression” column, type the expression. The expression syntax is not checked if you use this method.

For more information about expressions, see Chapter 23.

## Substituting tag names used in graphic objects

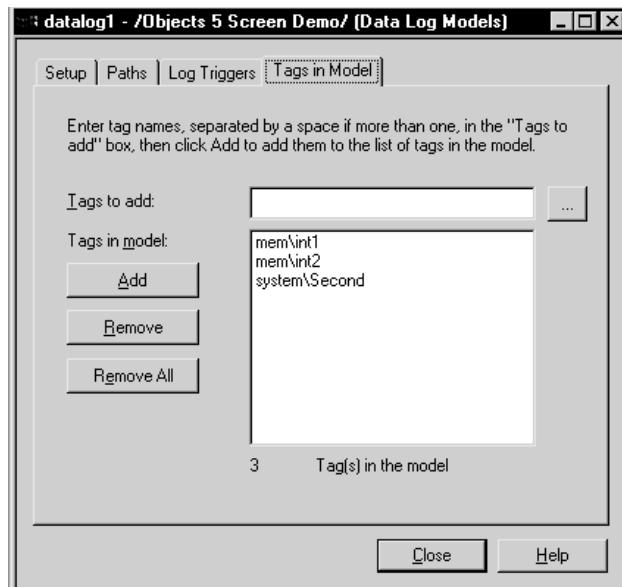
You can use tag substitution to replace tag names and expressions assigned to the connections for graphic objects. For example, if you assign a tag to multiple objects in a graphic display, and then decide to use a different name for the tag, you can create a new tag and then use tag substitution to search for all tag references with the old name, and replace them with the new tag name.

You can also use tag substitution to replace the tags contained in embedded variables.

For more information about tag substitution, see page 20-38.

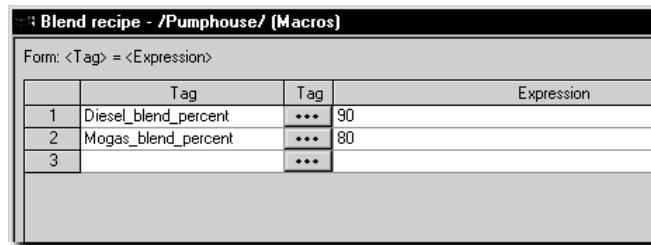
## Logging tag values

Analog and digital tag values can be logged, including both HMI and data server tags. To set up data logging, use the Data Log Models editor. For more information, see Chapter 26.



## Using macros to assign values to tags

A macro is a list of tag assignments stored in a text file. Whenever you run the macro, the specified values are written to the tags. For more information about macros, see Chapter 30.



You can also use recipes to assign values to multiple tags with a button press. For information about using recipes to assign values to tags, see Chapter 29.



# Using HMI tags

This chapter describes:

- HMI tag types.
- using the Tags editor.
- data sources.
- addressing syntax for device tags.
- organizing HMI tags.
- viewing tag statistics.
- other methods for creating HMI tags.
- importing HMI tags.

## HMI tag types

HMI tags are tags you create in FactoryTalk® View. For information about when to use HMI tags versus when to use data server tags, see Chapter 6.

You can create and use these types of HMI tags in FactoryTalk View:

<b>This tag type</b>	<b>Stores this type of data</b>
Analog	Range of values. Use analog tags to represent variable states such as temperature or the position of rotary controls.
Digital	0 or 1. Use digital tags to represent devices that can only be on or off, such as switches, contacts, and relays.
String	ASCII string, series of characters, or whole words (maximum of 82 characters). Use string tags to represent devices that use text, such as a bar code scanner that uses an alphanumeric product code.

This tag type	Stores this type of data
System	Information generated while the system is running, such as the name of the logged-in user, system time and date, and so on.  FactoryTalk View creates system tags when you create an application. The system tags are stored in the folder called "system" in the HMI tag database. You cannot edit system tags. You can use system tags anywhere you would use any other type of tag. For a list of system tags, see Appendix C.

## Analog tags that use floating-point values

You can assign analog tags to most of the connections in FactoryTalk View, including both HMI and data server tags. If the analog tag uses a floating-point data type but an integer value is required, the floating-point value is rounded.

For information about the data types available for analog tags, see Help.

### How values are rounded

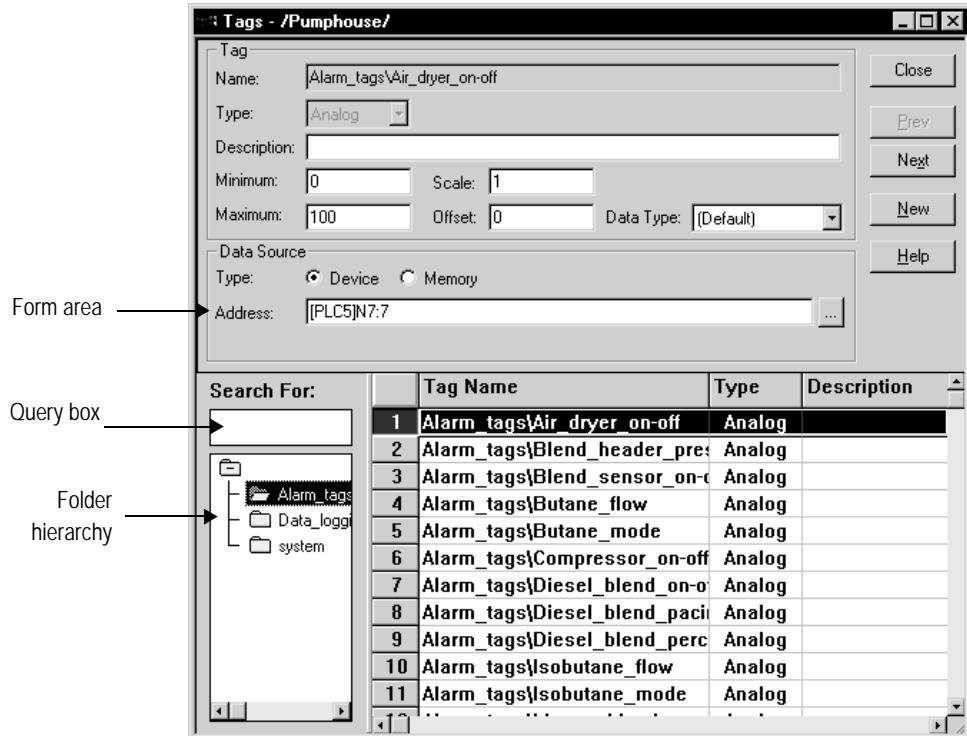
When a floating-point value must be rounded to an integer, this is how the value is rounded:

- If the number after the decimal is 4 or less, the value is rounded down. For example, 8.495 is rounded to 8.
- If the number after the decimal is 6 or more, the value is rounded up. For example, 8.6 is rounded to 9.
- If the number after the decimal is 5, the value is rounded using a "banker's algorithm," to average out the rounding over time:
  - If the number before the decimal is 0, the value is rounded down. For example, 10.5 is rounded to 10.
  - If the number before the decimal is an odd number, the value is rounded up. For example, 11.5 is rounded to 12.
  - If the number before the decimal is an even number, the value is rounded down. For example, 12.5 is rounded to 12.
- Floating point values are rounded to a maximum of six digits. Therefore, using a floating-point data type and adding to it a value that exceeds the six significant digits will result in a rounding error. For example, adding any number to 999,999 will result in a rounding error.

This rounding method is also used for numeric input enable buttons that use the implicit decimal position, if an integer tag is assigned to the button's Value connection. For more information about the numeric input enable button, see page 21-30.

## Using the Tags editor

The Tags editor has these parts: form area, query box, folder hierarchy, and spreadsheet area.



The information you enter in the form area is displayed in the spreadsheet area. The folder hierarchy displays all the folders that you and FactoryTalk View create.

The Tags editor has special items on the View menu to control the appearance of the editor, and extra tools on the toolbar.

Use the Tags editor to:

- create folders
- duplicate, nest, and delete folders

- create and view tags
- edit, duplicate, and delete tags

For details about using the Tags editor, see Help.

## Updating the contents of the Tags editor

You can create tags in the Tags editor, but you can also create them as you work in other editors, and you can import them from an existing tag database. To see the result of changes to the database you make in other editors while the Tags editor is open, update the contents.

### To update the contents of the editor



Refresh tool

1. On the View menu, click Refresh, or click the Refresh tool.

When you click Refresh, the undo buffer is cleared and recent changes can't be automatically reversed.

Refresh is not available if you haven't accepted the current tag.

For information about creating tags in other editors, see page 7-8.

## Searching for HMI tags

Use the Tags editor's query box to select the tags you want to display in the spreadsheet. This allows you to edit tags in different folders without browsing the folder hierarchy. You can:

- select a single tag by typing the tag name.
- select multiple tags by typing wildcard characters.

These are the wildcard characters:

This character	Does this
?	Matches any single character.
*	Matches any number of characters, including the backslash (\) character. Use this wildcard character by itself to display all the tags in the HMI tag database.

For example, to search for all the tags that contain the string "pump" in their name, type \*pump\* in the Search For box. When you press Enter, a list of tags containing the string "pump" appears in the spreadsheet.

When you do searches, remember that the backslash in a folder name counts as a character in the tag name.

## Data sources

When creating an HMI tag, you must specify a data source. The data source determines whether the tag receives its values externally or internally.

### The data source

The FactoryTalk View documentation uses the term data source as a generic term that includes all possible sources of tag data, for both data server tags and HMI tags. The data source can be memory or a device such as a programmable controller or an OPC® server. FactoryTalk View writes values to and reads values from the data source. The data source is set up to exchange information (in the form of numeric or string values) between the FactoryTalk View value table and the machine that your application is controlling.

#### Device

A tag with device as its data source receives its data from a source external to FactoryTalk View. The data can come from a programmable controller or other device via RSLink® Enterprise™, or from another OPC server. You can use up to 5,000 device tags.

For analog tags with Device as the data source, the data type determines the format used when reading and writing tag data to the data server.

#### Memory

A tag with memory as its data source receives its data from (and stores values in) the FactoryTalk View internal value table. Tags with memory as the data source do not count toward the total tag limit.

For analog tags with Memory as the data source, select the data type that matches the format of the data you will store in the tag. Choosing Default is the same as choosing Floating Point.

## Addressing syntax for device tags

For device tags, you must provide the address and path to the OPC server that is connected to the data source. For example, the OPC server could be connected to a device such as a programmable controller.

The syntax for OPC tag addresses is:

[AccessPath]Address

or

::[AccessPath]Address

where :: indicates that the address resides in a data server in the application. (The two colons are necessary when the address contains a colon, for example, N7:0.)



The square brackets are part of the syntax. They do not indicate optional parameters.

The access path is one of the following:

- For communications with an RSLinx Enterprise data server, the access path is the name of the device shortcut in RSLinx Enterprise.
- For communications with an RSLinx® Classic™ OPC server, the access path is the name of an OPC topic in RSLinx Classic.
- For communications with other OPC servers, the access path might be optional. For information about the syntax for the access path, see your OPC server documentation.

---

### Example: Logix5000 addressing

We recommend that you use the Tag Browser to select OPC items. When you use the Tag Browser the correct syntax and data types are selected automatically.

However, if you want to define tags before setting up communications, you can type the address manually, as shown in this example. If you type the address manually, be sure to select the data type for the tag that matches the data type used by the Logix5000 processor. If you use the Default data type, values are treated as floating-point values.

This example shows you how to specify the OPC topic, item name, and bit level of an item that uses the integer data type in a Logix5000 processor.

#### OPC topic and item addressing

Logix5000 uses this addressing syntax:

*[Device shortcut]item name* for RSLinx Enterprise.

*[OPC topic name]item name* for RSLinx Classic.

In this example, the RSLinx Enterprise device shortcut name for the Logix5000 processor is CLGX. To connect to an OPC item called Motor, type this address:

[CLGX]Motor

#### Bit level addressing

You can gain access to the bit level of an item that uses the integer data type by appending .# to the end of the tag address, where # is the number of the bit you want access to.

For example, to gain access to the first bit of the integer item Motor, type this address:

[CLGX]Motor.0

---

## Organizing HMI tags

Organizing tags makes finding and using tags faster and simpler. To organize your tags, try these tips:

- Establish naming conventions.

Naming conventions enable you to use wildcard characters most effectively when searching for and displaying tags during development.

- Use folders to group related tags.

## Naming tags

Tag names can be up to 255 characters long including folder name. If you create a folder, the folder name becomes part of the tag name. The backslash (\) after a folder name counts as a character.

The tag name can contain these characters:

- A to Z
- 0 to 9
- underscore (\_) and dash (-)

Tag names cannot contain spaces. The tag name can be mixed case. Tag names preserve upper and lower case for readability but are not case sensitive. For example, the tag name MixerValve1 is the same as mixervalve1.

When a tag name starts with a number or contains a dash, enclose the name in curly brackets { } when you use it in an expression, for example, {N33-0}. For more information about using tags in expressions, see Chapter 23.

## Using folders to group tags

To organize tags, create a folder to store tags that are related to one another. To separate the folder name from the rest of the tag name, use a backslash (\). For example, tags in the folder called Pump would start with Pump\.

For greater organization, you can nest folders. For example, you can organize the HMI tag database first by area, then by machines in the area, and finally by devices in each machine. The result might be Area1\Machine1\Pump\.

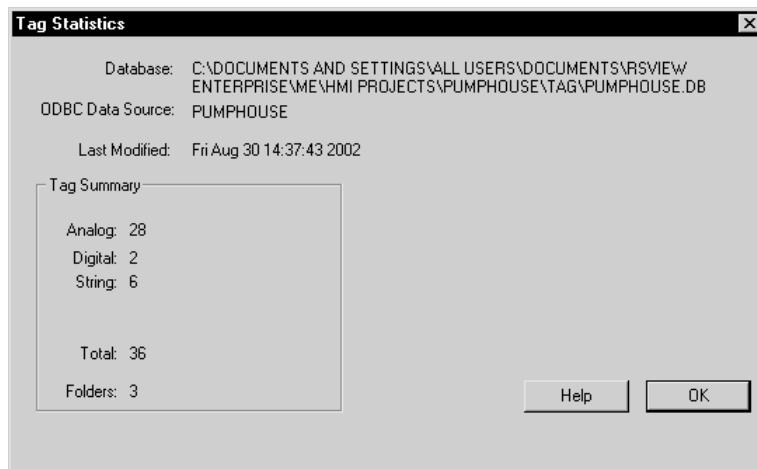
To create similar groups of tags quickly, create one folder with its tags and then duplicate the folder.

## Viewing tag statistics

The Tag Statistics dialog box provides a summary of how many tags your application uses, as well as other tag information such as the date the HMI tag database was last modified. To view tag statistics, the Tags editor must be open and have focus.

### To view tag statistics

1. On the View menu, click Tag Statistics.



## Other methods for creating HMI tags

In addition to creating tags in the Tags editor, you can:

- create tags as needed in other editors (using the Tag Browser).
- import tags from a PLC or SLC™ database. For details see page 7-9.

## Creating tags as needed in other FactoryTalk View editors

You can create tags as needed in any editor or dialog box that uses tags. You can add a tag to the database by doing one of the following:

- Click the Browse button in the Tag column to open the Tag Browser and create the tag. For information about using the Tag Browser, see page 6-5.
- Type a new tag name in the "Tag or expression" column. Keep track of the tag name, and add the tag in the Tags editor when convenient.



Browse button in the  
Tag column



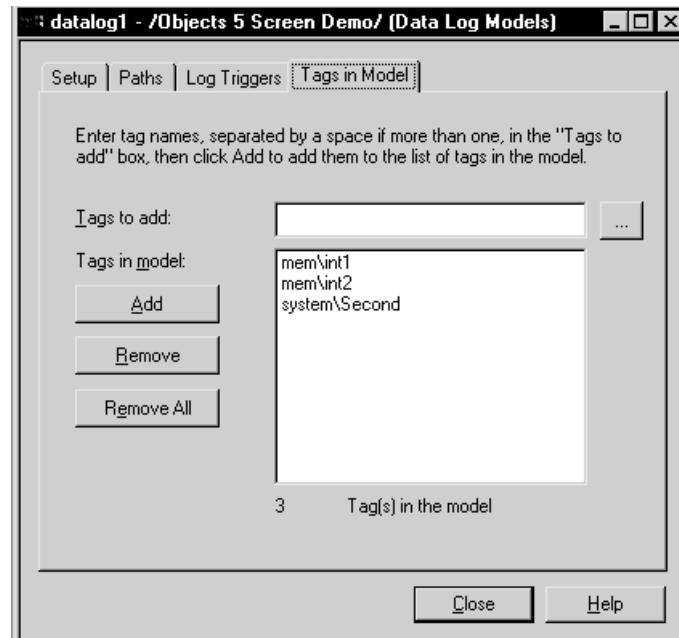
Browse button in the Data Log Models editor

## Creating tags as needed in the Data Log Models editor

In the Data Log Models editor, in the Tags in Model tab, you can add a tag to the database by doing one of the following:

- Click the Browse button beside the “Tags to add” box to open the Tag Browser and create the tag. For information about using the Tag Browser, see page 6-5.
- Type a new tag name in the “Tags to add” box, and then click Add. You are warned that the tag does not exist. Click Yes to add the tag to the list of tags in the datalog model. Keep track of the tag name, and add the tag in the Tags editor when convenient.

For more information about the Data Log Models editor, see Chapter 26.



## Importing tags from a PLC database

Use the Import PLC Tags dialog box to selectively import tags from a PLC or other database into your application’s HMI tag database. Tags imported in this way are copied into the database—they are not shared with the source database. This means changes to tags in your application do not affect the database from which they have been imported, and vice versa.

You can import tags from any of these databases:

- legacy PLC databases, created using WINtelligent™ LOGIC 5 or A.I.™ 5, with file extension .dsc
- RSLogix™ 5 or RSLogix 500, saved as an external database, with file extension .ctd (before importing, explicitly export to the .ctd format to be sure of including the latest changes)
- RSLogix 5 internal database, with file extension .rsp
- RSLogix 500 internal database, with file extension .rss

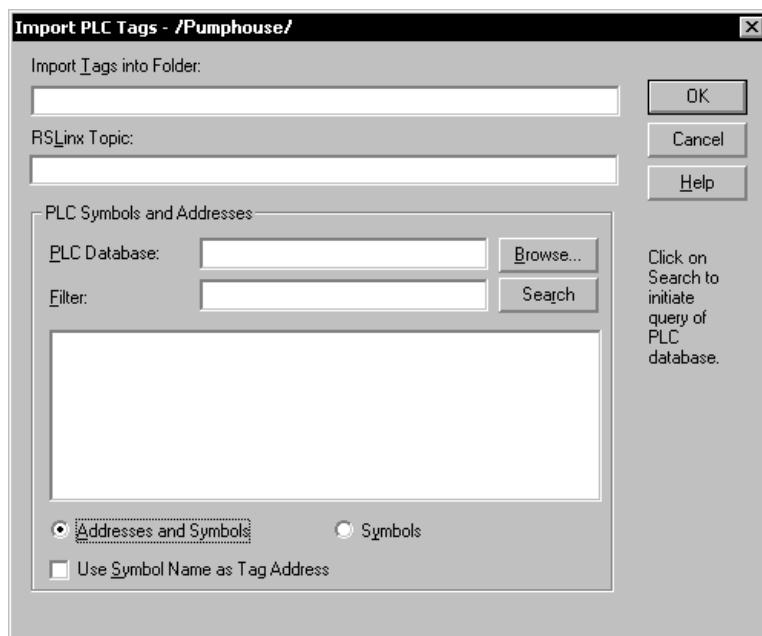
For PLC and SLC addresses, the Import PLC Tags dialog box shows only addresses that are used in the symbol or address list of the PLC programming software.



DB Browser tool in  
the Tags editor

### To open the Import PLC Tags dialog box, do one of the following

- In the Tag Browser, right-click the right pane, and then click Import PLC Tags. For information about using the Tag Browser, see page 6-5.
- In the Tags editor, on the Edit menu click Other Databases, or click the DB Browser tool.



For more information about using the Import PLC Tags dialog box, see Help.

## Using the Tag Import and Export Wizard

You can also import PLC or SLC databases into FactoryTalk View using the Tag Import and Export Wizard.

You can use the wizard to:

- export your application's tags to a .csv file.
- import tags from another FactoryTalk View application's tag .csv file.
- merge tags from another FactoryTalk View application (that is, import them directly from the application, without first creating a tag .csv file).
- import tags from legacy PLC databases, created using WINtelligent LOGIC 5 or A.I. 5, with file extension .dsc.
- import tags from RSLogix 5 or RSLogix 500, saved as an external database, with file extension .ctd (before importing, explicitly export to the .ctd format to be sure of including the latest changes).

### To start the wizard, do one of the following

- In FactoryTalk View Studio, on the Tools menu, click Tag Import and Export Wizard.
- On the Windows® Start menu, select Programs, Rockwell Software, FactoryTalk View, Tools, and then click Tag Import and Export Wizard.

For details about using the wizard, see the wizard's Help.





# 8 Setting up global connections

This chapter describes:

- what global connections are.
- updating the date and time.
- changing displays.
- controlling display changes remotely.
- printing displays.
- running macros.
- setting up the backlight intensity remotely.
- monitoring runtime RAM usage.

## About global connections

Global connections are connections that apply to your entire runtime application. Global connections allow the data source to control or interact with your application at run time.

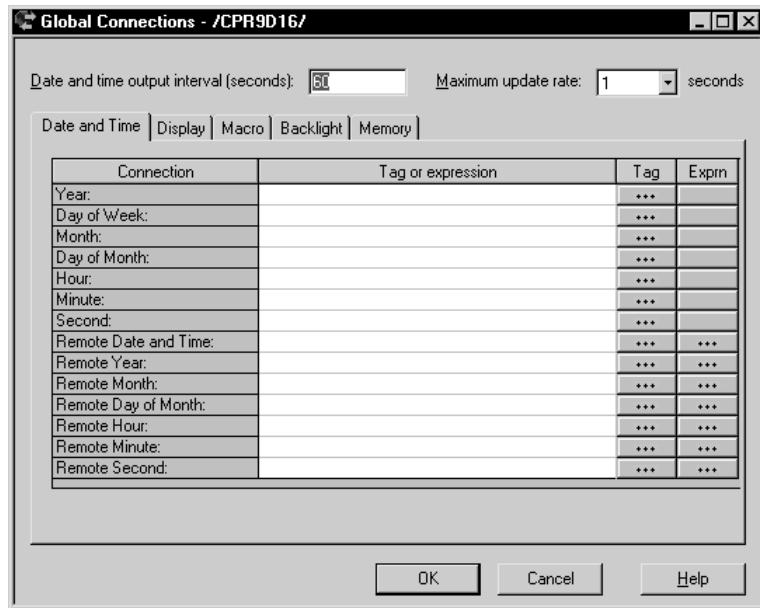
You can use global connections to:

- update the date and time at the data source using the runtime terminal's date and time.
- update the date and time on the runtime terminal using the data source date and time.
- notify the data source of the current display number.
- notify the data source when a display is printed.
- change the display on the runtime terminal remotely (from the data source).
- print the current display from the data source.
- run up to five macros from the data source, when a tag or expression result changes.
- set the intensity of the backlight on a PanelView™ Plus terminal.
- monitor runtime RAM usage for troubleshooting purposes.

The use of each global connection is optional.

## Setting up global connections

Use the Global Connections editor to set up global connections.



For detailed information about the options in the Global Connections editor, see Help.

### Updating the date and time

Use the date and time connections in the Date and Time tab of the Global Connections editor to update the date and time. You can assign any or all of the date and time connections, or leave them unassigned.

Normally you would assign only one set of date and time connections, either to update the date and time at the data source from the terminal, or to update the terminal's internal clock from the data source.

The values of assigned date and time connections are updated in the order in which the connections are listed in the Global Connections editor.

## Updating the date and time at the data source from the terminal

To update the date or time at the data source from the terminal's internal clock, assign a tag to one or more of these connections:

This connection	Contains this range of values
Year	00-99
Day of Week	1-7, with 1 being Sunday
Month	1-12
Day of Month	1-31
Hour	0-23
Minute	0-59
Second	0-59

Specify how frequently to update the values of the assigned connections at the data source (from the terminal's internal clock) by typing a time in the "Date and time output interval" box.

## Updating the date and time at the terminal from the data source

This option is especially useful if you have several terminals connected to the same data source. Instead of the operator resetting the date and time manually on each terminal, the data source can reset all internal clocks at once.

To update the date or time in the terminal's internal clock from the data source, assign a tag or an expression to the Remote Date and Time connection. Also assign a tag or an expression to one or more of these connections:

This connection	Contains this range of values
Remote Year	00-99 or 0000-9999
Remote Month	1-12
Remote Day of Month	1-31
Remote Hour	0-23
Remote Minute	0-59
Remote Second	0-59

When the Remote Date and Time connection value changes to a new non-zero value, the terminal resets its internal time and date to the time and date read from the individual remote date and time connections.

The date and time are updated at the rate specified in the “Maximum update rate” box, but changes are not made until the Remote Date and Time connection changes to a new non-zero value.

## Changing displays

Use these global connections in the Display tab of the Global Connections editor to monitor and control display changes:

- Display Number—notifies the data source of the number of the display that's currently open.
- Remote Display Number—allows the data source to change the display on the runtime computer.

## Controlling display changes remotely

To control display changes remotely, you can set up the data source to open graphic displays.

The data source can control display changes for Replace displays only. You can set up the data source to control all Replace display changes, or to open displays under set conditions, allowing the operator to change displays the rest of the time.

To use remote display changes, assign a unique display number to each Replace display, and assign a tag or expression to the Remote Display Number connection. When the data source sends a display's number to the connection, the specified display opens. When the connection's value is 0, the operator has control of display changes (using display navigation objects).

For information about assigning numbers to displays, see Help.

## Remote display changes and security

If you set up the data source to open graphic displays remotely, remote display changes occur whether or not the logged-in user has security access to a given display.

## Setting up remote display changes

This section outlines the steps for setting up remote display changes. For more detailed information about the Global Connections editor, see Help.

## To set up remote display changes

1. In the Graphics editor, in the Display Settings dialog box, assign a unique display number to each Replace display.

By default, every Replace display is assigned the number 1. Therefore, to use remote display changes you must assign a unique display number to every Replace display, even if you don't plan to open a particular Replace display remotely.

Another option is to avoid using the display number 1 for remote display changes. Then you only need to assign display numbers to the displays you want to change remotely.

For more information about graphic displays, see Chapter 19.

2. In the Global Connections editor, assign a tag or an expression to the Remote Display Number connection.
3. If you assign a tag to the Remote Display Number connection, set up the data source to send a display's number to the connection when you want the display to open.

If you assign an expression to the Remote Display Number connection, FactoryTalk® View monitors the values of the tags used in the expression, and uses the expression result to determine which display to open.

## Printing displays

Use these global connections in the Display tab of the Global Connections editor to monitor and control display printing:

- Display Print Active—notifies the data source that a display print is in progress.
- Remote Display Print—allows the data source to trigger a display print.

The operator can also print the current display by pressing the display print button. For information about setting up this button, see Help.

When the data source or the operator prints a display, everything on the screen is printed, including the current display, pop-up windows, and any visible background applications.



If you plan to print graphic displays remotely, turn off the screen saver. If the screen saver is on when a remote display print is triggered, only the screen saver image is printed.

For information about turning off the screen saver on personal computers, see your Windows® documentation.

For information about turning off the screen saver on a PanelView™ Plus or PanelView Plus CE terminal, see the *PanelView Plus Terminals User Manual*.

## Running macros

Use these global connections in the Macro tab of the Global Connections editor to run macros:

This connection	Runs this macro
Remote Macro1	Macro1
Remote Macro2	Macro2
Remote Macro3	Macro3
Remote Macro4	Macro4
Remote Macro5	Macro5

Create the macros in the Macros editor. Make sure you name them exactly as shown in the right column of the preceding table (with no space in the name).

When the tag or expression assigned to one of these connections changes to a new non-zero value, FactoryTalk View runs the associated macro.



Leave time for a macro to run before starting a new macro. Otherwise, FactoryTalk View will queue the macros, which consumes memory and could lead to system instability. When you shut down FactoryTalk View ME Station, if a message about executing macros is displayed for more than a few seconds this means that macros have been queued.

For information about creating macros, see Chapter 30.

## Setting up backlight intensity remotely

For applications that will run on PanelView Plus terminals, use the Remote Backlight Intensity connection in the Backlight tab of the Global Connections editor to set the backlight intensity of a runtime terminal remotely from a data source.

To update the backlight intensity of the runtime terminal from the data source, assign a tag or an expression to this read-only connection.

This connection	Contains this range of values
Remote Backlight Intensity	0-100

At run time, the application reads the value at the data source and sets the backlight to the percentage of its maximum intensity corresponding to the value. For instance, if the value

of the connection is 25, the backlight will be set to 25% of its maximum intensity. This value will override the terminal's own backlight intensity setting.

If the value of the connection is greater than 100, the backlight will be set to its maximum intensity. If the value is less than 0, the backlight is set to its minimum intensity. That is, it will be turned off.

## Monitoring runtime RAM usage

Sometimes it is helpful to monitor runtime RAM usage for troubleshooting purposes. If your application seems to be running slowly, consuming too much memory, or interfering with the operation of other applications on the same personal computer or terminal, it is useful to monitor the exact amount of memory the applications are consuming.



The PanelView Plus and PanelView Plus CE terminals' operating systems have a 32 MB RAM process limit. On these terminals, FactoryTalk View ME Station has a tag connection threshold of 26 MB. Screen change performance will slow down significantly as the system adds and removes tag connections from the cache. Use the runtime RAM usage global connections to monitor runtime RAM usage.

To reduce runtime RAM usage, reduce the number of displays or tags in your application, and restart the application regularly (for example, once a week). Adding more RAM to the terminal will not help.

For more information, see Answer ID 39481 in the Rockwell Automation Knowledgebase.

Use these global connections in the Memory tab of the Global Connections editor to monitor runtime RAM usage:

This connection	Provides this information, in kilobytes (Kb)
MERuntime RAM Usage	The amount of memory being used by your application (and by the MERuntime.exe process)
Total RAM Usage	The total amount of memory being used by all applications (and processes) on the personal computer or terminal
Available RAM	The remaining amount of memory available
Remote RAM Usage	Assign a tag or expression to this connection. When it has a non-zero value, data is sent to the other three connections, if assigned.

You can monitor the values of the MERuntime RAM Usage, Total RAM Usage, and Available RAM connections using data logging or using graphic objects such as numeric displays. The connections are updated every 10 seconds as long as the Remote RAM Usage connection has a non-zero value. Set up the data source to set this connection to a non-zero value when you want memory monitoring to occur, or assign the same tag or

expression to a graphic object such as a maintained push button that the operator can use to turn memory monitoring on and off.

These connections are available when you run FactoryTalk® View ME Station on a personal computer and on the PanelView Plus and PanelView Plus CE terminals.

For PanelView Plus and PanelView Plus CE terminals, you can also turn on the display of RAM usage statistics at run time. To do this, in the Configuration Mode window select Runtime RAM Usage, and then make your selections in the Runtime RAM Usage window. For more information, see the *PanelView Plus Terminals User Manual*. This manual is available on the FactoryTalk View Machine Edition CD.

 9

# Setting up alarms

This chapter describes:

- steps for setting up alarms.
- preparing to set up alarms.
- how alarms work.
- alarm trigger data types.
- tips for using array tags.
- optional alarm connections.
- creating alarm messages in multiple languages.
- the [ALARM], [ALARM BANNER], [ALARM MULTI-LINE], [STATUS], and [HISTORY] displays.
- using displays from the library in your application.
- creating your own alarm display.
- opening and closing the alarm display.
- how the alarm list, alarm banner, and alarm status list graphic objects work.
- using buttons to acknowledge, silence, clear, and delete alarms.
- using buttons to sort alarms and reset alarm status.

## About alarms

An alarm occurs when something goes wrong or is about to go wrong. Alarms can signal that a device or process has ceased operating within acceptable, predefined limits, and can indicate breakdown, wear, or process malfunctions. Alarms are also used to indicate the approach of a dangerous condition.

Alarms are an important part of most plant control applications because an operator must know the instant something goes wrong. It is often equally important to have a record of the alarm and whether the alarm was acknowledged.

## Multiple language alarm messages

A key component of any alarm system is the messages that appear when alarms occur. FactoryTalk® View supports alarm messages in multiple languages. For information about setting up alarm messages in multiple languages, see page 9-17.

## Other multiple language alarm features

FactoryTalk View supports using multiple languages for all captions on alarm objects. In addition, these alarm features can be translated into multiple languages:

- title of the alarm history report
- title of the alarm status report

Alarm objects and reports use the time formats for the current application language. For more information about using multiple languages, see Chapter 12.

## Summary of steps

Follow these steps to set up alarms:

1. In the Alarm Setup editor, set up alarm triggers (the tags or expressions to monitor), define the alarm messages and their trigger values, and specify the graphic display to open when alarms occur (if any).

Also use this editor to specify trigger types, “Acknowledge all” values, the maximum alarm log file size, the hold time, and optional connections.

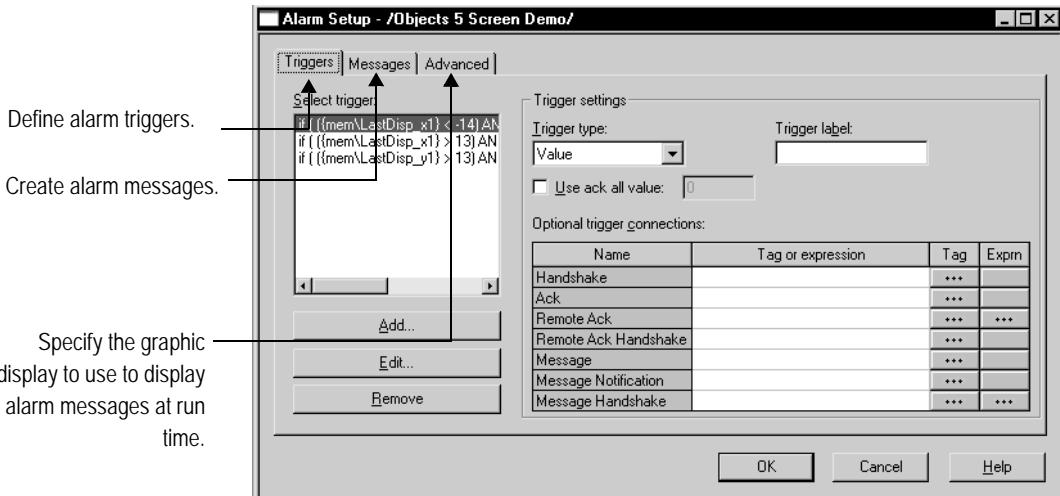
2. In the Startup editor, ensure that the Alarms box is checked (it is checked by default). See Help.
3. If desired, in the Graphics editor modify the default [ALARM] display, or create your own graphic display to use for alarms. For example, if you won’t be using audible alarm signals, edit the default display to remove the silence alarms button.

For information about graphic displays, see Chapter 19.

4. Test alarms on the runtime system.

## Setting up alarms

Use the Alarm Setup editor to set up alarms.



For detailed information about the options in the Alarm Setup editor, see Help.

### Preparing to set up alarms

As your application is running, information is continually sent to the data source about the state of the various processes. For example, your application might be monitoring whether a valve is open or closed, or the temperature in a boiler. Values representing the status of these processes are sent to the data source.

The first step in setting up alarms is determining which processes to monitor for alarm conditions.

### The data source

The FactoryTalk® View documentation uses the term data source as a generic term that includes all possible sources of tag data, for both data server tags and HMI tags. The data source can be memory or a device such as a programmable controller or an OPC® server. FactoryTalk View writes values to and reads values from the data source. The data source is set up to exchange information (in the form of numeric or string values) between FactoryTalk View and the machine that your application is controlling.

### Tags and expressions

Before setting up alarms, you must set up data server or HMI tags corresponding to the addresses at the data source that will store the values you want to monitor for alarm

conditions. You can monitor analog and digital tags for alarm conditions, including both HMI and data server tags. You cannot monitor string tags.

For information about setting up data server tags, see your OPC data server documentation. For information about creating HMI tags, see Chapter 7.

You can also use expressions to perform logical or mathematical calculations on tag values, and then monitor the expression value rather than the original tag value. For example, you could use an expression to monitor whether a tag value has increased or decreased beyond a threshold value: If Tag1 > 90 then 1 else 2.

For information about creating expressions, see Chapter 23.

### **Identifying alarm conditions**

Once you have identified the processes you want to monitor for alarms, and the tags or expressions that will store the values representing the status of the processes, you must determine the acceptable range of values for each tag or expression. Then you can set up alarms to notify the operator when a value is outside the normal operating range.

### **Importing and exporting alarm setup files**

The Alarm Import Export Wizard in FactoryTalk View Studio allows you to export alarm information to an XML file, and to import an alarm setup XML file.

For example, you can export an application's alarm setup information, import the setup to another FactoryTalk View application, and then modify it in the new application to suit your requirements. Or, you can modify the file before importing it into the new application.

You can use the Alarm Import Export Wizard to import alarm setup information that has been created using an external programming tool or editor, or you can import a FactoryTalk View XML file.

For more information about importing and exporting alarm setup files, see Appendix E.

## **How alarms work**

This section provides an overview of the key components of your application's alarm system, and describes how the different parts work together.

### **Alarm triggers and trigger values**

You specify the tags and expressions (also known as connections) to monitor for alarm conditions by creating an alarm trigger for each connection.

Each alarm trigger can generate one or more alarm messages, associated with different trigger connection values. For each alarm trigger, you specify the trigger values that will generate alarm messages, and create the messages to display for the trigger values.

The trigger value can be a non-zero integer value or a bit position, depending on which trigger data type you assign. For more information about trigger data types, see page 9-11.

## Filtering alarm triggers in multiple languages

FactoryTalk View 5.00 supports alarm triggers in multiple languages. When you create an alarm trigger, its name is in the current application language. You can export alarm triggers for translation and then import them back into the application. For details, see Chapter 12.

Various graphic objects allow you to specify the alarm triggers to which the object's action applies, using the name of the alarm trigger. This is called filtering alarms. When you translate trigger names into other languages, the default language is used to display the trigger name in the object's selection list. These graphic objects allow you to specify alarm triggers using a selection list called the Trigger Label Selector:

- alarm list
- alarm status list
- alarm banner
- clear alarm history button
- acknowledge all alarms button
- print alarm history button
- print alarm status button

If the trigger label name has not been set up in the default language, the Trigger Label Selector displays a question mark (?) instead of the trigger label name. To correct the display you must import the trigger labels for the default language.

You can also manually type in the trigger label names to use to filter alarms. If you manually type in the trigger names, specify the names in the default language.

If you change the default language, you must re-specify the trigger labels for each of the graphic objects listed above, using the trigger label names in the new default language. For more information about the default language, see page 12-2.

## Alarm notification methods

You can use any combination of these methods to notify the operator that an alarm condition has occurred:

- Open an alarm graphic display containing an appropriate alarm message.
- Set off an audible signal (if the application is running on a personal computer).

- Send a message to a printer.
- Send a message to the data source.

 If you set up alarm messages in multiple languages, make sure that the data source can either receive the message in Unicode format or convert it into ASCII characters for all the languages you will be using.

To use these methods, select the Display, Audio, Print, and Message to Tag options when you create your messages.

 Be sure to select the Print option if you need to keep a permanent record of alarms. You can also print the contents of the alarm log file periodically, as described on page 9-10. (The alarm log file is also referred to as the alarm history.)

## Displaying alarm information

FactoryTalk View comes with these alarm displays:

- [ALARM], in the Displays folder, contains an alarm banner graphic object that is set up to display the newest active alarm in the alarm log file.
- [ALARM BANNER], a display in the Libraries folder, contains an alarm banner graphic object that is set up to display the newest active alarm in the alarm log file.
- [ALARM MULTI-LINE], a display in the Libraries folder, contains an alarm list graphic object that is set up to display all the alarms in the alarm log file: active, inactive, acknowledged, and unacknowledged.
- [STATUS], a display in the Libraries folder, contains an alarm status list graphic object that is set up to display active alarms. It contains a goto display button for opening the [HISTORY] display. This display is available in two sizes.
- [HISTORY], a display in the Libraries folder, contains an alarm list graphic object and a goto display button for opening the [STATUS] display. The alarm list is set up to display all the alarms in the alarm log file: active, inactive, acknowledged, and unacknowledged. This display is available in two sizes.

By default, the [ALARM] graphic display opens automatically at run time when an alarm is generated.

You can accept the default, edit the [ALARM] display, use one of the other alarm displays, or create your own display. If you elect not to automatically open a graphic

display for alarm notification, an alternative is to provide the operator with a way to open an alarm display when desired.

For more information about	See
The [ALARM] display	page 9-25
The [ALARM BANNER] display	page 9-26
The [ALARM MULTI-LINE] display	page 9-27
The [STATUS] display	page 9-28
The [HISTORY] display	page 9-29
Creating your own alarm display	page 9-30
The alarm list graphic object	page 9-32
The alarm banner graphic object	page 9-33
The alarm status list graphic object	page 9-34

## Interacting with alarms

Depending on which alarm objects you set up, the operator can:

- acknowledge one or all alarms, or the alarms for a specific alarm trigger.
- clear and delete alarms.
- silence alarms (if the application is running on a personal computer).
- sort alarms.
- reset the status of all alarms.
- print alarms.

For information about the alarm button graphic objects the operator can use to work with alarms, see page 9-35.

## Using the data source to interact with alarms

You can also set up remote acknowledgement, silencing, and resetting of alarms, to allow the data source to perform these functions. And you can set up the data source to determine which types of alarms to display (active, inactive, acknowledged, and unacknowledged), and to close the alarm display. These topics are discussed later in the chapter.

## Methods for acknowledging alarms

A single alarm is acknowledged when:

- the operator selects an alarm in the alarm list and presses the acknowledge alarm button, an enter button object, or the Enter key on an external keyboard or keypad.
- the alarm banner has focus and the operator presses the acknowledge alarm button.

A single alarm for a particular trigger is acknowledged when:

- the trigger's Remote Ack connection value changes to the alarm's trigger value (or bit position, for bit triggers).

All alarms for a particular trigger are acknowledged when:

- the trigger's Remote Ack connection value changes to the "Acknowledge all" value. The "Acknowledge all" value is specified in the "Use ack all value" box in the Alarm Setup editor.
- the operator presses an acknowledge all alarms button that is set up to acknowledge the alarms for a specific alarm trigger.

All alarms in the alarm log file are acknowledged when:

- the operator presses an acknowledge all alarms button that is set up to acknowledge all alarms.
- the Remote Ack All connection's value changes to a new non-zero value.

For more information about acknowledging alarms, see page 9-38.



The operator can send button presses to objects that do not have focus. For information about linking buttons to objects, see page 21-9.

## Methods for clearing and deleting alarms

Alarms are deleted from the alarm log file and cleared from all alarm lists and alarm banners when the operator presses the clear alarm history button. You can set up the button to clear alarms for all alarms, or for a specific alarm trigger.

The alarm in the alarm banner is cleared when the operator presses the clear alarm banner button. It is not deleted from the alarm log file.

For more information about clearing and deleting alarms, see page 9-39.

## Methods for silencing alarms

The audio alarm indicator is available for applications running on personal computers only.

All alarms are silenced when:

- the operator presses the silence alarms button.
- an alarm whose Audio property is turned on is acknowledged.
- the Remote Silence connection's value changes to a new non-zero value.
- the operator presses the clear alarm history button.

### **Method for sorting alarms**

When the operator presses the sort alarms button, the sort order toggles from time to trigger value or from trigger value to time. The sort order applies to all alarms in all alarm lists and in the alarm log file.

### **Methods for resetting alarms**

Resetting alarms affects how alarms are displayed in the alarm status list. When alarms are reset:

- the accumulated time an alarm has been in alarm is reset to 0. If the alarm is still active, the time begins accumulating again.
- the number of times the alarm has been triggered is reset to 0. If the alarm is still active, the number changes to 1.
- the date and time are sent to the system tag AlarmResetDateAndTimeString.

The alarms for a specific alarm trigger are reset when:

- the operator presses a clear alarm history button that has been set up to clear and reset alarms for a specific alarm trigger.

All alarms are reset when:

- the operator presses a reset alarm status button.
- the operator presses a clear alarm history button, if the button has been set up to clear and reset all alarms.
- the Remote Status Reset connection's value changes to a new non-zero value.

For more information about resetting alarms, see page 9-40.

## Methods for printing alarm information

In addition to automatically printing each alarm message as it occurs, you can keep a printed record of alarms:

- using the print alarm history button. When the operator presses the button, a report is printed of the contents of the alarm log file. The report can include the time alarms occurred and were acknowledged.

You can set up this button to print the alarm history for a selected alarm trigger, or for all alarms.

If you set up alarm messages in multiple languages, messages and trigger labels are printed in the language that they were originally logged in. (Messages are logged in the application language that is current at the time the message is triggered.) All alarm times and acknowledge times are printed in the current application language.

- using the print alarm status button. When the operator presses the button, a report is printed of the status of alarms. The report can include active alarms, past alarms, or all alarms that have been set up in the Alarm Setup editor. The report can include how many times each alarm was triggered, and the accumulated time in alarm.

You can set up this button to print the alarm status for a selected alarm trigger, or for all alarms.

If you set up alarm messages in multiple languages, all messages, trigger labels, and times are printed in the current application language.

For information about specifying which printer to use at run time for applications that will run on a personal computer, see page 15-10. For information about the type of printer to use with a PanelView™ Plus or PanelView™ Plus CE terminal, see page 16-1.

## The alarm log file

As soon as the application starts running, FactoryTalk View begins monitoring trigger connections for alarm conditions. When an alarm occurs, FactoryTalk View adds the associated message to the alarm log file. When the file is full, the oldest messages are deleted to make room for new messages. Specify the maximum number of messages to store in the Alarm Setup editor.

For each message, the log file records the time the alarm was triggered and the time the alarm was acknowledged (if it was acknowledged).

The log file is saved every 30 seconds, if there are new alarms. The log file is also saved when the application is shut down.

The alarm log file is also referred to as the alarm history. For example, the operator presses the clear alarm history button to remove all alarms from the log file.

The alarm log file is retained when you restart an application after a shutdown or power loss. You can delete the log file from the runtime computer at application startup.

For information about deleting the log file, see page 15-15.

If you download a newer version of an application to the runtime computer, the alarm log file for the older version is deleted automatically.

## Alarm trigger data types

When you create alarm triggers, you must specify the type of data the trigger's tag or expression uses. The type of data affects how alarms are triggered.

### The Value trigger type

To trigger alarms based on integer or floating-point values, use the Value trigger type.

When the value of the trigger connection equals the trigger value of a message, an alarm is generated.

If you use an analog tag (either an HMI tag or a data server tag) or an expression, you can use any non-zero integer or floating-point value to trigger an alarm. Floating-point values are rounded to the nearest integer. For information about how values are rounded, see page 7-2.

Trigger values cannot be 0. Digital tags have two possible values, 0 and 1. Therefore, if you use a digital tag (either an HMI tag or a data server tag) you can only use the value 1 to trigger a message. If you want to use a digital tag to trigger two different messages, create an expression that adds 1 to the digital tag's value. That way, you can use the trigger values 1 and 2.

---

### Example: Integer trigger values

This example shows how to set up alarms using integer trigger values to monitor the level of acid in a 100-gallon tank. Alarms are generated when the tank is 75% full and 90% full.

1. Create an alarm trigger for the tag Acid\_tank\_level. This tag points to an address in a programmable controller that is linked to a sensor in the tank. The tag's minimum value is 0 and its maximum value is 100.

Use this expression for the alarm trigger:

```
if ((Acid_tank_level > 74) AND (Acid_tank_level < 90))
```

```
then 1
```

```
else if (Acid_tank_level > 89)
```

```
then 2
```

else 0

2. Use the Value trigger type for the alarm trigger.
3. Specify these trigger values and alarm messages for the alarm trigger:

Trigger Value	Message
1	Warning! The acid tank has exceeded safe fill levels.
2	Danger! The level in the acid tank is too high.

At run time, when the value of Acid\_tank\_level is 75 or greater, FactoryTalk View generates the first alarm message. The second message is generated when the tag value is 90 or greater.

Using the expression in step 1 allows a range of values to trigger each alarm. This is useful in case the tag is not being read at the exact moment it reaches a threshold level.

---

## The Bit trigger type

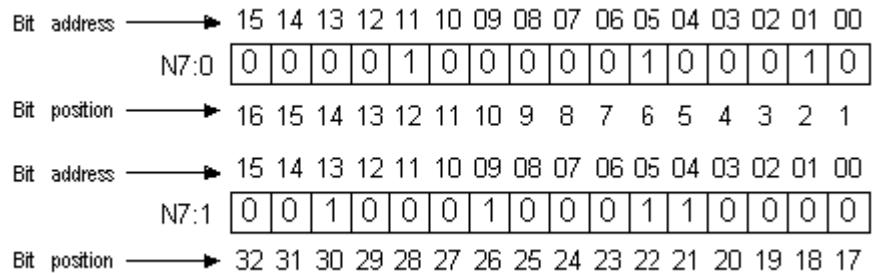
Use the Bit trigger type to generate multiple simultaneous alarm messages. You can assign an array tag (consisting of up to 1,024 bits) to the trigger connection. Each bit in the array whose value changes from 0 to 1 triggers an alarm (if a message is set up for the bit position).

---

### Example: Trigger values using bit positions

This example shows how to use bit positions to generate multiple simultaneous alarms. The array tag uses 32 bits, from N7:0/00 to N7:1/15.

This diagram shows the bit addresses and bit positions for the array tag:



The tag is currently generating alarms at bit positions 2, 6, 12, 21, 22, 26, and 30.

### To set up alarms for this array tag:

1. Create an alarm trigger for the array tag. Use the Bit trigger type.

You must use a direct reference for the array tag. The tag uses the Integer data type (16 bits per element). Append L2 to the tag's starting address to indicate that the trigger will monitor 2 tag elements (32 bits), as follows:

{::[PLC5]N7:0,L2}

PLC5 is the RSLinx® Enterprise™ device shortcut name. The two colons at the beginning are necessary when the tag reference contains a colon.

2. Specify trigger values and alarm messages for the alarm trigger. Each trigger value corresponds to a bit position (not a bit address).

Bit in the array (bit address)	Bit position	Trigger value	Message
00	1	1	Line 1: Conveyor has stopped.
01	2	2	Line 1: Power failure.
02	3	3	Line 2: Conveyor has stopped.
03	4	4	Line 2: Power failure.
..	..	..	..
31	32	32	Oven door open.

### The Least Significant Bit (LSBit) trigger type

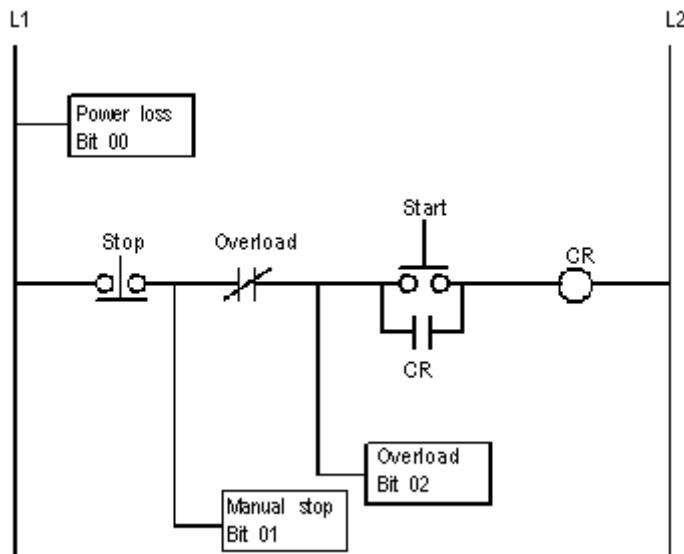
Use the Least Significant Bit trigger type to generate alarm messages based on a priority sequence that is determined by bit position. You can assign an array tag (consisting of up to 1,024 bits) to the trigger connection. When multiple bits in the array change from 0 to 1, only the alarm with the lowest bit position is generated.

---

### Example: Trigger values using least significant bits

This example shows how to use the Least Significant Bit trigger type to prioritize which alarm is generated when multiple alarm conditions occur.

In this example, a programmable controller monitors a motor's power, manual stop, and overload status. The diagram indicates the location of sensors that are linked to an array tag in the programmable controller.



If power to the motor is lost, the manual stop and overload switch also lose power, generating alarms for all three motor parts. The only alarm the operator needs to see is the alarm for the motor, since the cause of the alarm is power loss to the motor, not a problem with the other two motor parts.

If someone stops the motor manually, power to the overload switch is lost. Alarms are generated for the manual stop and the overload switch, but the only alarm the operator needs to see is the alarm for the manual stop.

1. Create an alarm trigger for the tag `Motor_starter`. The tag uses 16 bits, and points to the programmable controller address `N7:61`, although only the first three bit positions are used in this example. Use the `LSBit` (Least Significant Bit) trigger type.

Because you are using only the first three bits in a single tag element, there is no need to specify a length for the array tag in this example. You can use a simple tag type.

However, if you were also monitoring bits in `N7:62`, you would need to add `L2` to the address, as follows:

```
{::[PLC5]N7:61,L2}
```

2. Specify these trigger values and alarm messages for the alarm trigger:

Bit in the array (bit address)	Bit position	Trigger value	Message
00	1	1	The motor has lost power.
01	2	2	The motor has stopped.
02	3	3	The overload switch has lost power.

At run time, if power to the motor is lost, the programmable controller changes all three bit values from 0 to 1, but FactoryTalk View generates the first alarm message only (since bit 00 is the lowest—that is, least significant—bit in alarm). If the operator acknowledges the first alarm and power is not yet restored, the second alarm is generated, and so on.

---

## Tips for using array tags

You can use these tag data types for array tags:

Data type	Number of bits
Byte	8
Integer	16
Word	16
Long Integer	32
Double Word	32

The right column shows the number of bits for each data type. This is useful for helping to determine how many data elements to address to achieve the required number of bits. To create an array tag, specify how many elements to use, beginning with the first element at the tag's address.

For example, if you want to monitor 128 bit positions for an alarm trigger, and you are using the Long Integer data type, you would append the modifier [4] to the tag name (because  $4 \times 32 = 128$ ). For detailed information about tag syntax, see page 9-16.

You cannot use HMI tags for array tags. You must use data server tags for array tags.

## Equivalent data types

Different processors and OPC servers use different terminology for tag data types. This section provides information about equivalent data types that you can use for array tags, with the associated number of bits for each.

### PLC and SLC tags

When creating array tags, you can use PLC and SLC tags with this data type:

Data type	Number of bits
Short Integer	16

Data server tags with the Short Integer data type are displayed in the Tag Browser with the Item Canonical Data Type of Integer.

### To view a tag's Item Canonical Data Type

1. In the Tag Browser, right-click the tag, and then click Properties.

### Logix5000 tags

When creating array tags, you can use Logix5000 tags with these data types:

Data type	Number of bits
SINT	8
INT	16
DINT	32

Data server tags with the data types described in the previous table are displayed in the Tag Browser with the Item Canonical Data Types of Character, Short Integer, and Long Integer, respectively.

### RSLinx Enterprise tag syntax

To create an RSLinx Enterprise array tag, use this syntax to specify the length of the array:

*{tagname,Larraylength}*

For example, `{::[PLC5]N7:0,L5}`

The tag N7:0 uses the Short Integer data type (16 bits), so the array tag consists of 80 bits (16 x 5).



When the tag name contains a colon (:), put two colons before the RSlinx Enterprise device shortcut name.

### **KEPServerEnterprise tag syntax**

To create a KEPServerEnterprise array tag, use this syntax to specify the length of the array:

*tagname [arraylength]*

For example, Chan1.Dev1.Tag1 [3]

The tag Tag1 uses the Long Integer data type (32 bits), so the array tag consists of 96 bits (32 x 3).

Alternatively, you can define the tag length in KEPServerEnterprise when you create the tag, and then just select the tag name in FactoryTalk View Studio. For more information, see KEPServerEnterprise Help.

### **Creating alarm messages in multiple languages**

FactoryTalk View 5.00 supports alarm messages in multiple languages. When you create alarm messages, they are in the current application language. You can export the alarm messages for translation and then import them back into the application. For details, see Chapter 12.

### **Language switching alarm messages in RSView ME Station 4.00**

For applications that will run in RSView ME Station version 4.00, use the CurrentLanguage( ) expression function to specify offsets for each trigger's messages in the Alarm Setup editor. In the editor, divide the messages for each trigger into sections for each language. For information about the CurrentLanguage( ) function, see page 23-15.

### **Optional alarm connections**

When an operator acknowledges an alarm, the acknowledge time is recorded in the alarm log file and the alarm is silenced. This might be all you require of your alarm system.

However, you can use the optional alarm connections in the Alarm Setup editor to set up more sophisticated interactions with the data source when alarms are triggered, acknowledged, silenced, and reset. For example, you can set up the data source to acknowledge, silence, and reset alarms. You can also use optional alarm connections to close the alarm display or to allow the data source to close the alarm display.

There are two sets of optional alarm connections:

- connections that work with a specific alarm trigger. You can assign a different set of connections to each alarm trigger.

- connections that apply to all alarms

You can also assign connections to alarm lists, to allow the data source to determine which types of alarms to display in the lists. For more information, see page 9-32.

## Connections that work with a specific alarm trigger

You can assign a different set of these connections to each alarm trigger, or to only one or some alarm triggers:

- Handshake—Assign a tag to this connection to notify the data source when the trigger's value changes.
- Ack—Assign a tag to this connection to notify the data source when the operator acknowledges an alarm (or all alarms).
- Remote Ack—Assign a tag or an expression to this connection to allow the data source to acknowledge alarms.
- Remote Ack Handshake—Assign a tag to this connection to notify the data source when a remote acknowledgement occurs.
- Message—Assign a tag to this connection to send alarm messages to the connection. Make sure the tag supports the type of data sent in the alarm message. For example, if the message is a string of text, assign a string tag to the Message connection.
- Message Notification—Assign a tag to this connection to notify FactoryTalk View that an alarm message has been sent to the connection and FactoryTalk View must wait before sending a new message.
- Message Handshake—Assign a tag to this connection if you want the data source to notify FactoryTalk View when it has read the message.

Assign tags or expressions to these connections when you create your alarm triggers. For more detailed information about how these connections work, see the next sections.

## How the Handshake connection works

The Handshake connection is useful for triggers that use the Value data type. Program the data source to queue alarm notifications when multiple alarms are generated for the same trigger. Use the Handshake connection to notify the data source that FactoryTalk View has detected the alarm notification. Then the data source can send the next alarm to FactoryTalk View.

When the application starts, the Handshake connection has a value of 1. When the trigger connection's value changes, the Handshake connection's value toggles from 1 to 0. The next time the trigger value changes, the Handshake connection's value toggles from 0 to 1.

A change in the trigger connection's value does not necessarily indicate an alarm—the value might still be within acceptable limits.

## How the Ack connection works

When the application starts, this connection has a value of 0.

### How the value changes

- When the operator acknowledges an alarm by pressing the acknowledge alarm button, FactoryTalk View sends the alarm's trigger value (or bit position, for bit triggers) to this connection, and holds the value as long as the operator presses the button, or for the duration of the hold time, whichever is longer. Then the connection's value is reset to 0.
- When the operator acknowledges all alarms, or the alarms for this trigger only, by pressing the acknowledge all alarms button, FactoryTalk View sends the trigger's "Acknowledge all" value to this connection, and holds the value as long as the operator presses the button, or for the duration of the hold time, whichever is longer. Then the connection's value is reset to 0.
- When the Remote Ack connection changes to a new non-zero value that matches the trigger value of an unacknowledged alarm, FactoryTalk View sends the trigger's value (or bit position, for bit triggers) to this connection, and holds the value for the duration of the hold time. Then the connection's value is reset to 0.

To avoid confusion about whether a specific trigger's alarm individual alarm has been acknowledged or all alarms for the trigger have been acknowledged, make sure the "Acknowledge all" value is not the same as one of the trigger values.

## How the Remote Ack connection works

The Remote Ack connection allows for the remote acknowledgement of alarms. The data source writes the alarm's trigger value (or bit position, for bit triggers) to this connection, and the unacknowledged alarm with this trigger value is acknowledged.

If you assign this connection and specify an "Acknowledge all" value for the trigger, when the data source sends the "Acknowledge all" value to this connection, all unacknowledged alarms for the trigger are acknowledged.

Make sure that the "Acknowledge all" value is not the same as one of the trigger values.

Set up the data source to send the correct value to the Remote Ack connection when you want a remote acknowledgement to take place.

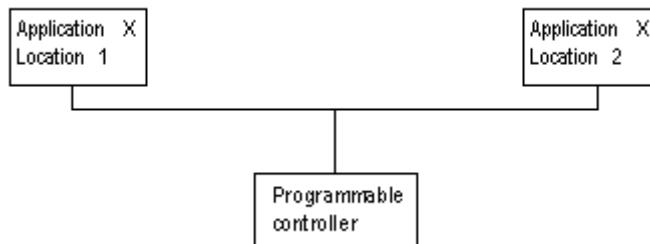
Remote acknowledgement does not take place if the hold time is in effect for the Ack connection.

---

### Example: Using the Remote Ack connection

This example shows how to use the Remote Ack connection to acknowledge an alarm.

In this example, the same application is run at two different locations on the plant floor, and both terminals are connected to the same programmable controller. All the tags in the example refer to addresses in the programmable controller.



1. Create an alarm trigger for a tag called Alarm. Use the Value trigger type.
2. Assign a tag called Ack to the Ack connection.
3. Assign a tag called Remote\_ack to the Remote Ack connection.
4. Create an alarm message for the alarm trigger, with the trigger value 1.
5. Set up the data source to write the value at the Ack connection to the Remote Ack connection whenever the value of the Ack connection changes from 0 to a non-zero value.

At run time, when an alarm with the trigger value 1 is generated, the alarm message appears at both locations simultaneously. If the operator acknowledges the alarm at Location 1, FactoryTalk View writes the value 1 to the Ack connection.

Since the Ack connection is write-only (not read), the acknowledgement doesn't appear at Location 2. However, the programmable controller writes the value from the Ack connection (1) to the Remote Ack connection, and the alarm at Location 2 is acknowledged.

---

### How the Remote Ack Handshake connection works

This connection is used to notify the data source that the remote acknowledgement has been detected by FactoryTalk View.

When the application starts, the Remote Ack Handshake connection has a value of 1.

When the Remote Ack connection value changes, the Remote Ack Handshake connection's value toggles from 1 to 0. The next time the Remote Ack value changes, the Remote Ack Handshake value toggles from 0 to 1.

The Remote Ack Handshake value toggles whether or not the new Remote Ack value matches a trigger value.

## Ensuring alarm messages are read by the data source before sending new messages

This section explains how the Message, Message Notification, and Message Handshake connections work.

To ensure an alarm message is read by the data source before FactoryTalk View sends a new message, use message handshaking. While message handshaking is in effect for an alarm message, FactoryTalk View cannot send a new message to the alarm trigger's Message connection.

### Methods of alarm message handshaking

Alarm handshaking works by setting the alarm trigger's Message Notification connection to 1. As long as the Message Notification connection is set to 1, new messages cannot be sent to the Messages connection. How the Message Notification connection is reset to 0 depends on how you set up alarm message handshaking.

There are two ways you can use alarm message handshaking:

- to hold the message at the data source for a specific period of time.
- to hold the message at the data source until the data source notifies FactoryTalk View that the message has been read.

Choose the method that best suits your application needs and communication system.

### Holding the message for a specific period of time

To set up an alarm trigger's alarm message handshaking so that the message at the Message connection is held for a specific period of time, assign a tag to the Message Notification connection and specify a hold time in the Advanced tab of the Alarm Setup editor.

### How handshaking works

This method of alarm message handshaking works as follows:

1. When an alarm occurs, the alarm message is sent to the Message connection.
2. The Message Notification connection is set to 1.

As long as the Message Notification connection is set to 1, FactoryTalk View cannot send new messages to the data source.

3. The “hold time” timer begins timing.
4. When the hold time has expired, the Message Notification connection is reset to 0 and FactoryTalk View can send a new message to the Message connection.

### **Holding the message until the data source acknowledges that it has read the message**

To set up an alarm trigger’s alarm message handshaking so that the message at the Message connection is held until the data source notifies FactoryTalk View that it has read the message, use two connections: the Message Notification connection and the Message Handshake connection.

Set up the data source to send a new non-zero value to the Message Handshake connection when it has read the message at the Message connection.

### **How handshaking works**

If you use the Message Handshake connection, handshaking works like this:

1. When an alarm occurs, the alarm message is sent to the Message connection.
2. The Message Notification connection is set to 1.

As long as the Message Notification connection is set to 1, FactoryTalk View cannot send new messages to the data source.

3. When the data source has read the message, it sends a new non-zero value to the Message Handshake connection.
4. The Message Notification connection is reset to 0 and FactoryTalk View can send a new message to the Message connection.

### **How messages are queued**

FactoryTalk View can queue up to 128 messages while it waits for the Message Notification connection to be reset to 0. Queued messages are sent to the data source on a first in, first out basis.

If the alarm message queue fills before the Message Notification connection is reset to 0, an error message is sent to FactoryTalk® Diagnostics.

### **How the Message connection works**

This connection is used to send the message associated with a trigger value to the data source.

The message is sent only if the alarm message's Message to Tag option is selected.

If you are going to generate text alarm messages in multiple languages, make sure the tag assigned to this connection can accommodate the string length of the messages in each language. For information about using multiple languages, see Chapter 12.

## How the Message Notification connection works

This connection is used to notify FactoryTalk View that an alarm message has been sent to the data source and FactoryTalk View must wait to send another message.

When the application starts, the Message Notification connection is set to 0. When a message is written to the Message connection, the Message Notification connection value changes to 1. When the hold time expires, or the Message Handshake connection changes to a new non-zero value, the Message Notification connection value changes from 1 to 0.

Messages for an alarm trigger are queued if the Message Notification connection is set to 1.

## How the Message Handshake connection works

Use the Message Handshake connection to reset the Message Notification connection when the data source has read the alarm message.

Set up the data source to send a new non-zero value to the Message Handshake connection when it has read the message at the Message connection.

## Connections that apply to all alarms

These connections apply to all alarms:

- Silence—Assign a tag to this connection to notify the data source that all alarms have been silenced (for applications that will run on a personal computer only).
- Remote Silence—Assign a tag or an expression to this connection to allow the data source to silence all alarms (for applications that will run on a personal computer only).
- Remote Ack All—Assign a tag to this connection to allow the data source to acknowledge all alarms at once.
- Status Reset—Assign a tag to this connection to notify the data source that the alarm status has been reset.
- Remote Status Reset—Assign a tag or expression to this connection to allow the data source to reset the alarm status.
- Close Display—Assign a tag to this connection to notify the data source that the alarm display has closed.

- Remote Close Display—Assign a tag or expression to this connection to allow the data source to close the alarm display.

Assign tags or expressions to these connections in the Advanced tab of the Alarm Setup editor. For more detailed information about how these connections work, see the next sections.

### **How the Silence connection works**

When the application starts, this connection has a value of 0. When alarms are silenced, this connection is set to 1 for the hold time and the internal beeper is turned off. At the end of the hold time, the connection is reset to 0.

If the silence event is triggered by a button press, this connection is set to 1 for the hold time or for the duration of the button press, whichever is longer. At the end of this time, the connection is reset to 0.

If subsequent alarms (that have been set up to sound the internal beeper) are generated, the alarms sound the beeper regardless of the Silence connection value or hold time status.

While the hold time is in effect, new alarms cannot be silenced.

### **How the Remote Silence connection works**

When this connection changes to a new non-zero value, all alarms are silenced. Alarms can be silenced using other methods regardless of this connection's value.

Set up the data source to send a new non-zero value to the Remote Silence connection each time you want a remote silencing of alarms to take place.

### **How the Remote Ack All connection works**

When this connection changes to a new non-zero value, all unacknowledged alarms are acknowledged.

Set up the data source to send a new non-zero value to the Remote Ack All connection each time you want a remote acknowledgement of all alarms to take place.

### **How the Status Reset connection works**

When the application starts, this connection has a value of 0. If a reset alarm status or clear alarm history button is pressed, this connection is set to 1 for the hold time or for the duration of the button press, whichever is longer. At the end of this time, the connection is reset to 0.

The Status Reset connection is also set to 1 if the Remote Status Reset connection changes to a new non-zero value, and remains set for the hold time.

While the hold time is in effect, the alarm status cannot be reset again. The status of alarms keeps updating during the hold time.

### How the Remote Status Reset connection works

When this connection changes to a new non-zero value, the status of all alarms is reset, as described on page 9-9. Alarms can be reset using other methods regardless of this connection's value.

Set up the data source to send a new non-zero value to the Remote Status Reset connection each time you want a remote resetting of alarm status to take place.

### How the Close Display connection works

When the application starts, this connection has a value of 0. If the alarm display (assigned in the Alarm Setup editor) closes, this connection is set to 1 for the hold time. At the end of the hold time, the connection is reset to 0.

If subsequent alarms are generated, the alarm display opens again regardless of the Close Display connection value or hold time status.

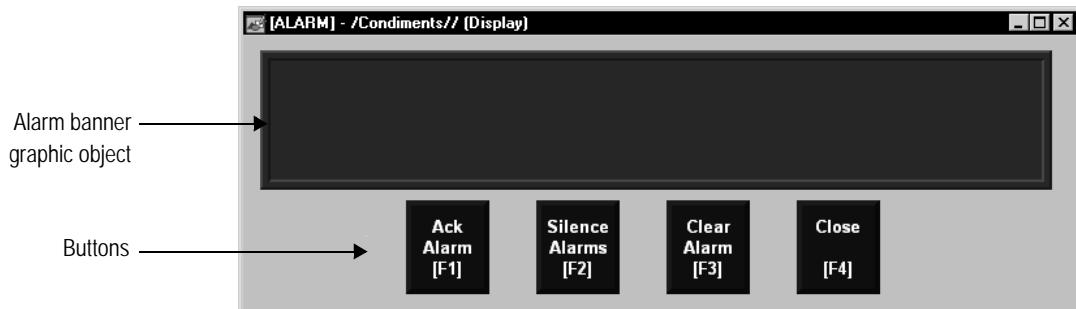
### How the Remote Close Display connection works

When this connection changes to a new non-zero value, the alarm display closes. The alarm display can be closed using other methods regardless of this connection's value.

Set up the data source to send a new non-zero value to the Remote Close Display connection each time you want a remote closing of the alarm display to take place.

## The [ALARM] display

When you create an application, it comes with a graphic display called [ALARM]. The [ALARM] display is the default display for showing alarm messages at run time. It contains an alarm banner graphic object that is set up to display a single active alarm.



You can use the [ALARM] display as is, or modify the display. For example, you can change the color of the objects, or add and remove buttons.

Another option is to create your own graphic display to use when an alarm occurs, as described on page 9-30. Or, you can use one of the alarm displays in the graphic library. These displays are described in the next four sections.

A copy of the [ALARM] display is included in the graphic library, with the name [ALARM BANNER].

### **The alarm banner graphic object**

The [ALARM] display contains an alarm banner graphic object, which lists the most recent alarm message. You can set up the alarm banner to queue new alarms until the displayed alarm has been acknowledged, or to show new alarms as they occur. The alarm banner in the [ALARM] display is set up to display new alarms as they occur and to display active alarms only (but you can edit it, if desired).

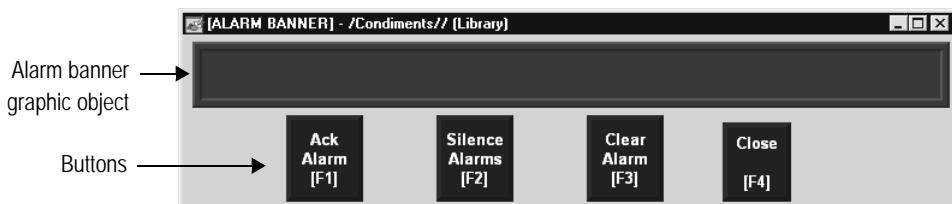
For more information about the alarm banner, see page 9-33.

### **Buttons in the [ALARM] display**

The [ALARM] display contains alarm buttons for acknowledging, silencing, and clearing alarms, as well as a button for closing the display. For information about how the buttons work, see page 9-35.

## **The [ALARM BANNER] display**

The Libraries folder contains a graphic display called [ALARM BANNER]. The [ALARM BANNER] display allows the operator to see a single alarm at a time.



### **The alarm banner graphic object**

The [ALARM BANNER] display contains an alarm banner graphic object, which lists the most recent alarm message. You can set up the alarm banner to queue new alarms until the displayed alarm has been acknowledged, or to show new alarms as they occur. The alarm banner in the [ALARM BANNER] display is set up to display new alarms as they occur and to display active alarms only (but you can edit it, if desired).

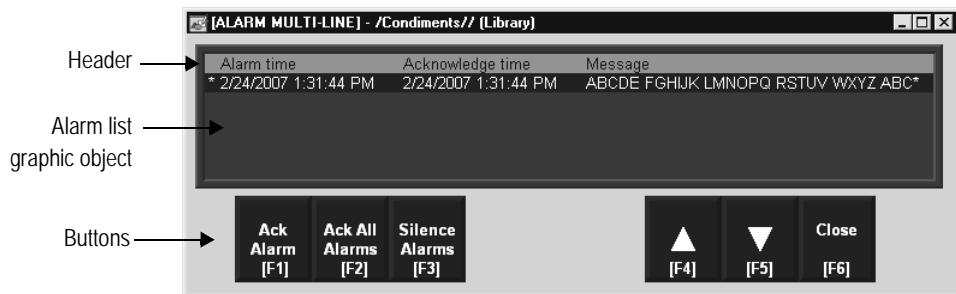
For more information about the alarm banner, see page 9-33.

### Buttons in the [ALARM BANNER] display

The [ALARM BANNER] display contains alarm buttons for acknowledging, clearing, and silencing alarms, as well as a button for closing the display. For information about how the buttons work, see page 9-35.

## The [ALARM MULTI-LINE] display

The Libraries folder contains a graphic display called [ALARM MULTI-LINE]. The [ALARM MULTI-LINE] display allows the operator to see multiple alarms at once.



You can use the [ALARM MULTI-LINE] display as is, or modify the display. For example, you can select which alarm triggers' alarms to display in the list.

### The alarm list graphic object

The [ALARM MULTI-LINE] display contains an alarm list graphic object, which lists the time the alarms occurred and were acknowledged, as well as the alarm messages. You can set up the alarm list to show any combination of active, inactive, acknowledged, and unacknowledged alarms. The alarm list in the [ALARM MULTI-LINE] display is set up to show all alarms (but you can edit it, if desired).

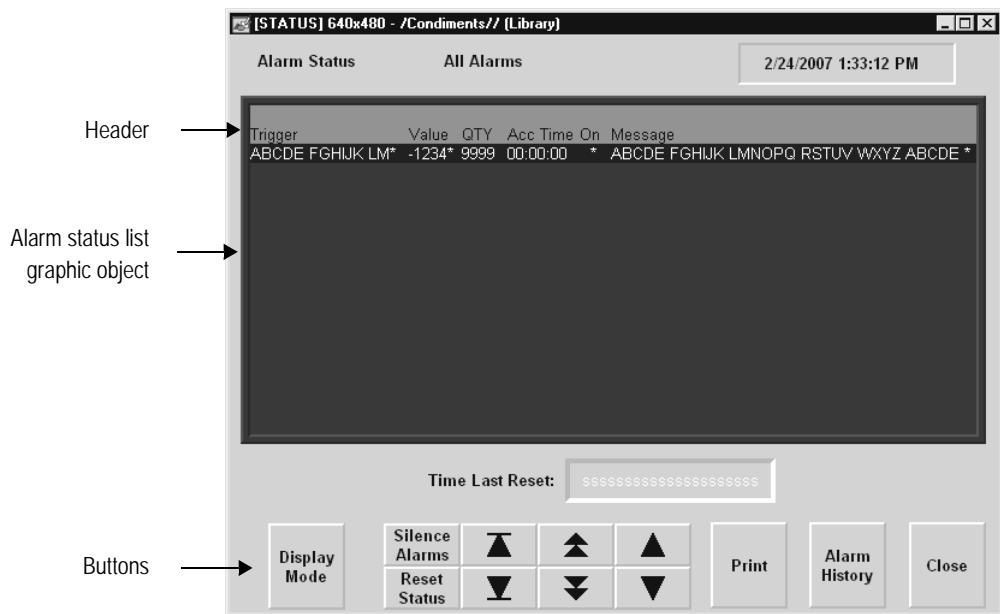
For more information about the alarm list, see page 9-32.

### Buttons in the [ALARM MULTI-LINE] display

The [ALARM MULTI-LINE] display contains alarm buttons for acknowledging and silencing alarms, as well as buttons for scrolling the list and closing the display. For information about how the buttons work, see page 9-35.

## The [STATUS] display

The Libraries folder contains a graphic display called [STATUS]. The [STATUS] display allows the operator to see the status of all the alarms that have been set up in the Alarm Setup editor. The [STATUS] display comes in two sizes, 640x480 and 800x600.



You can use the [STATUS] display as is, or modify the display. For example, you can select which alarm triggers' alarms to display in the list, or remove buttons that you don't want to use.

### The alarm status list graphic object

The [STATUS] display contains an alarm status list graphic object, which lists the alarm messages. You can set up the alarm status list to show the status of all alarms that have been set up, active alarms only, or all alarms that have occurred since the alarm status was last reset. The alarm status list in the [STATUS] display is set up to show the status of active alarms (but you can edit it, if desired).

For more information about the alarm status list, see page 9-34.

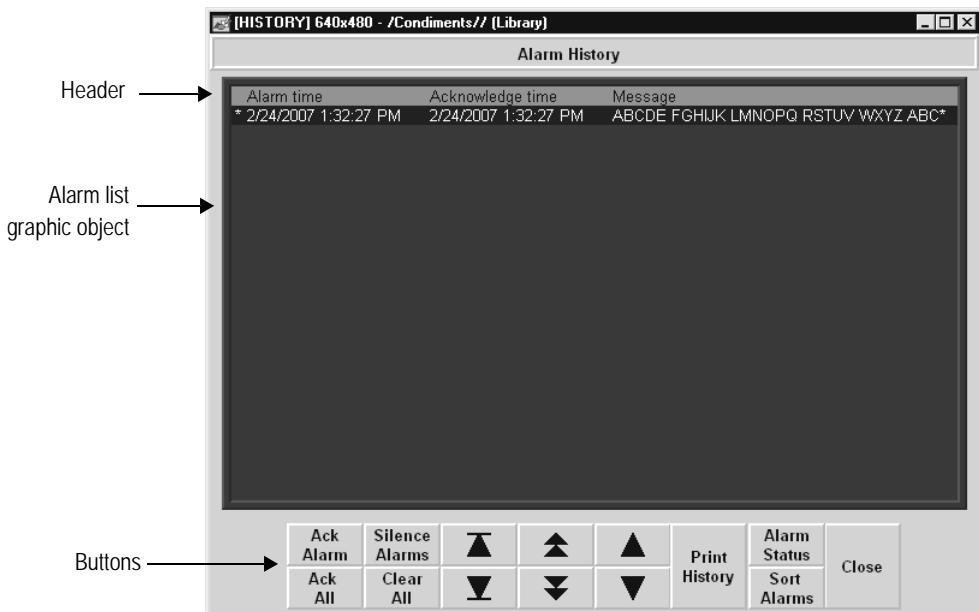
### Buttons in the [STATUS] display

The [STATUS] display contains alarm buttons for resetting and silencing alarms, changing the display mode, printing the status of alarms, and opening the [HISTORY]

display, as well as buttons for scrolling the list and closing the display. For information about how the buttons work, see page 9-35.

## The [HISTORY] display

The Libraries folder contains a graphic display called [HISTORY]. It is similar to the [ALARM MULTI-LINE] display, but the alarm list graphic object is larger (to display more alarm messages at once), and the [HISTORY] display contains more buttons. The [HISTORY] display comes in two sizes, 640x480 and 800x600.



You can use the [HISTORY] display as is, or modify the display. For example, you can select which alarm triggers' alarms to display in the list, or remove buttons that you don't want to use.

### The alarm list graphic object

The [HISTORY] display contains an alarm list graphic object, which lists the time the alarms occurred and were acknowledged, as well as the alarm messages. You can set up the alarm list to show any combination of active, inactive, acknowledged, and unacknowledged alarms. The alarm list in the [HISTORY] display is set up to show all alarms (but you can edit it, if desired).

For more information about the alarm list, see page 9-32.

### Buttons in the [HISTORY] display

The [HISTORY] display contains alarm buttons for acknowledging, silencing, and clearing alarms, sorting the list of alarms, printing the alarm history list, and opening the [STATUS] display, as well as buttons for scrolling the list and closing the display. For information about how the buttons work, see page 9-35.

### Using displays from the library in your application

The previous four sections described the [ALARM BANNER], [ALARM MULTI-LINE], [STATUS], and [HISTORY] displays that are included in the Libraries folder. You can copy objects from the displays to your own display, or you can copy the entire displays into your application.

For information about copying objects from a library into your graphic display, see page 20-42.

The graphic libraries are available on the development computer, but do not appear at run time. To use a display in the library as a graphic display at run time, you must add the library display into your application's folder of graphic displays. For details, see page 19-16.

---

### Example: Using the [ALARM] and [STATUS] displays together

To use the [ALARM] display to alert the operator about alarms as they occur, and the [STATUS] display to show how many times and for how long alarms have been active:

1. In the Alarm Setup editor, in the Advanced tab, select the [ALARM] display as the display to open when alarms are triggered, if it is not already selected. (It is selected by default.)
2. Add the [STATUS] display from the library.
3. Edit the [ALARM] display by adding a goto display button that opens the [STATUS] display when pressed.

---

### Creating your own alarm display

You can create your own graphic display for displaying and responding to alarms. For example, you could create a display with a static text object that notifies the operator that an alarm has occurred, and a goto display button that the operator can press to open a display that lists the actual alarm message.

If you create your own graphic display, use an On Top display and select the Cannot Be Replaced option.

For information about the graphic objects you can use to display alarm information, see the information starting on page 9-32. For information about creating graphic displays and graphic objects, see Chapter 19 and Chapter 20.

## Opening and closing the alarm display

### Opening the display

The alarm display you specify in the Advanced tab of the Alarm Setup editor (either the default [ALARM] display, a display you copy from the library, or your own display) is automatically opened whenever a trigger connection's value matches a trigger value (if you selected the Display option for the trigger value).

You can also create a goto display button that the operator can press to open the alarm display. For information about setting up a goto display button and specifying the display to open, see Help.

### Closing the display

The operator can close the display by pressing a close display button.

The display closes automatically under these circumstances:

- when the operator presses an acknowledge all alarms button and causes the newest alarm in the alarm history to be acknowledged.
- when the operator presses a clear alarm history button and causes the newest alarm in the alarm history to be deleted.
- when the Remote Close Display connection changes to a new non-zero value.
- when the Remote Ack All connection changes to a new non-zero value.
- when the newest alarm is acknowledged.

If you don't want the display to close when the newest alarm is acknowledged, you can turn off this option in FactoryTalk View ME Station.

### To change how the display closes

1. In FactoryTalk View ME Station, click Terminal Settings.
2. Double-click Alarms.
3. Specify how you want the display to close.

## How the alarm list graphic object works

The alarm list graphic object displays the time an alarm was triggered, and the time it was acknowledged (if you set up the object to display acknowledged alarms).

At run time, when a trigger connection at the data source matches a message's trigger value, an alarm appears in the alarm list. The alarm list can be in the [ALARM MULTILINE] or [HISTORY] display, in an alarm display you have created, or can be placed on any display in your application.

You can use multiple alarm lists, in the same display or in different displays. Each alarm list displays information from the same alarm log file, although you can set up different lists to display different information.

For information about setting up alarm lists, see Help.

## What is displayed

- If you set up the list to show the alarms for specific alarm triggers, only alarms for those triggers are displayed.
- If the list is set up to show unacknowledged alarms only, the alarms displayed in the list are unacknowledged and the acknowledge time column, if any, is always blank.
- If the list is set up to show both acknowledged and unacknowledged alarms, acknowledged alarms can show a symbol (which you can specify) at the left end of the row, and the acknowledge time appears in the acknowledge time column, if displayed.
- If the list is set up to show both active and inactive alarms, active alarms can show a symbol (which you can specify) at the left end of the row. If both the acknowledge symbol and active symbol are displayed, the active symbol column is to the right of the acknowledge symbol column.
- If the message is too long to fit in the object, the last displayed character is replaced with an asterisk. Similarly, if a column heading is too long to fit in the column, the last displayed character is replaced with an asterisk. You can specify how many lines to allow for each alarm message, from 1 to 10.
- The number of alarm messages that fits in the display depends on the height of the alarm list, the list's font size, how many lines you have specified for each alarm message, and whether the column headings are displayed.
- If alarms are currently being sorted by time, alarms are listed with the newest alarms on top.
- If alarms are currently being sorted by trigger value, alarms from the first trigger in the Alarm Setup editor are listed first, with alarms from subsequent triggers listed after.

Alarms in the list from the same trigger are grouped together and then sorted by trigger value, from lowest to highest.

- If there are multiple instances of the same alarm, these alarms are sorted by time with the newest on top.
- If you assign a tag or expression to any of the ActiveAcknowledged, ActiveUnacknowledged, InactiveAcknowledged, or InactiveUnacknowledged connections, when a connection's value is a non-zero value, the related type of alarm is displayed in the alarm list. The connection value overrides the setting for the list in the Alarm List Properties dialog box.
- If you set up alarm messages in multiple languages, messages are displayed in the language that they were originally logged in. All alarm and acknowledgement times are displayed in the current application language.

### How the list scrolls

- When a display containing an alarm list is opened, the highlight bar or cursor is at the top of the list: on the most recent alarm if sorted by time; or on the first alarm for the first trigger, if sorted by trigger value.
- If alarms are sorted by time, and the top alarm is selected, as new alarms occur, the old alarms scroll down, but the highlight bar or cursor remains at the top.
- If alarms are sorted by time, when the operator selects an alarm other than the top one in the list (by using the move down button), and new alarms occur, they are added above the visible area if the list is full. The currently selected alarm continues to be selected and maintains its position in the displayed portion of the list.
- If alarms are sorted by trigger value, as new alarms occur the currently selected alarm continues to be selected and maintains its position in the list.
- If the sort order changes, the selected alarm continues to be selected, but it might be displayed in a different position in the list.

### How the alarm banner graphic object works

The alarm banner graphic object displays a single, unacknowledged alarm.

At run time, when a trigger connection at the data source matches a message's trigger value, an alarm appears in the alarm banner. The alarm banner can be in the [ALARM] display or [ALARM BANNER] display, in an alarm display you have created, or can be placed on any display in your application.

You can use multiple alarm banners, in the same display or in different displays. You can set up different banners so that one displays the most recent alarm, and the other queues alarms until the displayed alarm is acknowledged.

For information about setting up alarm banners, see Help.

### **What is displayed**

- If you set up the banner to show the alarms for specific alarm triggers, only alarms for those triggers are displayed.
- You can set up the banner to queue new alarms until the operator clears the current alarm, or to always display the most recent alarm.
- When a display containing an alarm banner is first opened, the alarm banner is blank, unless the display is the alarm display specified in the Alarm Setup editor and it was opened in response to an alarm.
- When the displayed alarm is acknowledged, it is removed from the alarm banner. If there are no newer alarms, the banner is blank.
- If the operator presses a clear alarm banner button, the banner is blank.
- If the operator presses a clear alarm history button or if all alarms are acknowledged, the banner is blank.
- If the banner is set up to show both active and inactive alarms, when an alarm is active, an asterisk (\*) appears beside the message.
- If you set up alarm messages in multiple languages, messages and trigger labels are displayed in the language that they were originally logged in. All alarm times are displayed in the current application language.

### **How the alarm status list graphic object works**

The alarm status list graphic object displays the status of alarms, including whether an alarm has been triggered, how many times an alarm has been triggered, and for how long.

The alarm status list can be in the [STATUS] display, in an alarm display you have created, or can be placed on any display in your application.

You can use multiple alarm status lists, in the same display or in different displays. You can set up different lists to display different information.

For information about setting up alarm status lists, see Help.

### **What is displayed**

- If you set up the list to show the alarms for specific alarm triggers, only alarms for those triggers are displayed.
- If an alarm is active, an asterisk (\*) appears in the alarm state column, if displayed.

- If the message is too long to fit in the object, the last character that would fit is replaced with an asterisk. Similarly, if a column heading is too long to fit in the column, the last character that would fit is replaced with an asterisk. You can specify how many lines to allow for each alarm message, from 1 to 10.
- The number of alarm messages that fits in the display depends on the height of the alarm status list, the list's font size, how many lines you have specified for each alarm message, and whether the column headings are displayed.
- For value-triggered alarms, the accumulated time column shows how long the alarm has been set to the trigger value.
- For bit-triggered alarms, the accumulated time column shows how long the bit has been set to 1.
- For LSBit-triggered alarms, the accumulated time column shows how long the corresponding alarm's least significant bit has been set to 1.
- If you set up alarm messages in multiple languages, all messages and trigger labels are displayed in the current application language, regardless of what language they were originally logged in.

### What happens when the display is opened

- The first time a display containing the alarm status list is opened, the first page of alarms is displayed.
- If the list is set up to display active alarms only, and alarms that were visible when the display was closed are no longer active, the alarms are removed from the list.
- If alarms have been reset since the display was last opened, the first page of alarms is displayed.

## Using buttons with the alarm history and alarm objects

### Alarm buttons

You can use these alarm buttons to interact with the alarm list, alarm banner, or alarm status list, and to clear and sort the alarm history:

Use this button	With this graphic object	To do this
Acknowledge alarm	Alarm list Alarm banner	Acknowledge and silence the selected alarm.

Use this button	With this graphic object	To do this
Acknowledge all alarms		Acknowledge and silence all currently unacknowledged alarms, or the currently unacknowledged alarms for a specific alarm trigger.
Alarm status mode	Alarm status list	Change the type of alarms displayed in the alarm status list, from all alarms to active alarms to past alarms.
Clear alarm banner	Alarm banner	Clear the alarm in the alarm banner without removing the alarm from the alarm log file and alarm lists.
Clear alarm history		Remove alarms from the alarm log file and all alarm lists. If the Ack connection is assigned for any of the alarm triggers, the operator is prompted to acknowledge all alarms before clearing them.
		You can set up the button to remove all alarms, or just the alarms for a specific alarm trigger. You can also specify whether to reset the cleared alarms.
		If you choose to reset alarms, pressing this button resets the number of times an alarm has been triggered to 0, and the accumulated time in alarm to 0, for all alarms that are being cleared. If the alarm is still active, the number of times in alarm changes to 1 after the alarm is reset to 0.
Print alarm history		Print a report of the alarm messages in the alarm log file. You can include all alarms, or just the alarms for a specific alarm trigger. The report can include the time alarms occurred and were acknowledged.
Print alarm status		Print a report of the status of alarms. You can include all alarms, or just the alarms for a specific alarm trigger. The report can include how many times each alarm was triggered, and the accumulated time in alarm.

Use this button	With this graphic object	To do this
Reset alarm status		Reset the number of times an alarm has been triggered to 0, and the accumulated time in alarm to 0, for all alarms. If the alarm is still active, the number changes to 1.
Silence alarms		Silence the audio indicator for all alarms (for applications that will run on a personal computer only).
Sort alarms		Toggle between sorting alarms in alarm lists and the alarm log file by time and by trigger value.

You can assign any caption you choose to the labels on the alarm buttons.

### Linking buttons to objects

You can link the acknowledge alarm button to a specific alarm list or alarm banner, or set up the button to work with whichever alarm list or alarm banner is selected in the graphic display.

Similarly, you can link the clear alarm banner button to a specific alarm banner. This option is useful if you have multiple alarm banners in the same graphic display. And you can link the alarm status mode button to a specific alarm status list.

For more information about linking buttons to objects, see page 21-9.

### Key buttons

You can also use these key button graphic objects to interact with the alarm list, alarm status list, or alarm banner:

This button	Does this
Move up	Scrolls up one row in the list.
Move down	Scrolls down one row in the list.
Page up	Moves the highlight bar or cursor up one page in the list.
Page down	Moves the highlight bar or cursor down one page in the list.
Home	Moves the highlight bar or cursor to the top alarm in the list.

This button	Does this
End	Moves the highlight bar or cursor to the bottom alarm in the list.
Enter	Acknowledges the currently selected alarm (in alarm lists and alarm banners).

You can link key buttons to a specific alarm object, or set up the buttons to work with whichever object is selected in the graphic display. For more information, see page 21-9.

For information about creating graphic objects, see Chapter 20. For information about setting up key buttons, see Help.

## Using alarm buttons to acknowledge, silence, clear, and delete alarms

Acknowledging messages does not remove them from the alarm log file. You can still display acknowledged alarms in an alarm list that is set up to show acknowledged alarms.

### Acknowledging the selected alarm

When the operator presses the acknowledge alarm button, an enter button, or the Enter key on an external keyboard or keypad:

- the selected alarm in the alarm list is acknowledged and the audio indicator for the alarm, if any, is silenced.
- the displayed alarm in the alarm banner is acknowledged and the audio indicator for the alarm, if any, is silenced.

This is what happens when the alarm is acknowledged:

- If assigned, the alarm trigger's Ack connection is set to the alarm trigger value at the data source. The value is held as long as the operator presses the button, or for the hold time, whichever is longer.

If the hold time is still in effect and the operator acknowledges a new instance of the same alarm, the new acknowledgement is ignored.

- If the alarm list is set up to display unacknowledged alarms only, the alarm is removed from the list. The highlight bar or cursor moves up to the next unacknowledged alarm, unless it is already on the top unacknowledged alarm.
- If the highlight bar or cursor is on an acknowledged alarm when the operator presses the acknowledge alarm button, the button press is ignored.
- The alarm is removed from the alarm banner.

## Acknowledging all alarms

When the operator presses an acknowledge all alarms button, all unacknowledged alarms in the system (or, optionally, for a specific alarm trigger), are acknowledged.

This is what happens when the alarms are acknowledged:

- For each alarm trigger to which an “Acknowledge all” value is assigned, the specified “Acknowledge all” value is sent to the trigger’s Ack connection. The value is held as long as the operator presses the button, or for the hold time, whichever is longer. Then the connection’s value is reset to 0.

If the “Acknowledge all” value is not assigned for a trigger, no value is sent to the trigger’s Ack connection.

If the hold time for any alarm is still in effect and the operator presses acknowledge all alarms, the button press is ignored.

- If the alarm list is set up to display unacknowledged alarms only, all acknowledged alarms are removed from the list.
- The alarm banner is cleared.

## Silencing alarms

When the operator presses the silence alarms button, any audible signal triggered by an alarm is silenced and the Silence connection is set to 1 for the hold time or for the duration of the button press, whichever is longer. Then the connection’s value is reset to 0.

If the hold time for any alarm is still in effect and the operator presses silence alarms, the button press is ignored.

The audio alarm indicator is available for applications running on personal computers only.

## Clearing and deleting messages

### From the alarm history

When the operator presses the clear alarm history button, all alarms in the system (or, optionally, for a specific alarm trigger) are deleted from the alarm log file and from all alarm lists. For information about the alarm log file, see page 9-10.

If the Ack connection is assigned for any of the alarm triggers being cleared, the operator is prompted to acknowledge the unacknowledged messages before deleting them. If the operator chooses to acknowledge the alarms, the “Acknowledge all” value (if any) is sent to the Ack connection before deleting the messages. If the Ack connection is not assigned, or if the operator chooses not to acknowledge alarms, all messages are deleted immediately.

## From the alarm banner

### Using alarm buttons to sort alarms and reset alarm status

#### Sorting alarms

When the operator presses the sort alarms button, the sort order toggles from time to trigger value or from trigger value to time. The sort order applies to all alarms in all alarm lists and in the alarm log file.

#### Resetting alarm status

When the operator presses the reset alarm status button, the alarm status of each alarm is reset as follows:

- The accumulated time an alarm has been in alarm is reset to 0. If the alarm is still active, the time begins accumulating again.
- The number of times the alarm has been triggered is reset to 0. If the alarm is still active, the number changes to 1.

If the Status Reset connection is assigned (in the Alarm Setup editor), it is set to 1 for the hold time or for the duration of the button press, whichever is longer. If a hold time from a previous reset alarm status button press is still in effect, the button press is ignored.

Pressing the clear alarm history button can also reset the status of alarms, if you select the Reset alarm status option for the button. You can set the button up to clear and reset the status of all alarms, or just the alarms for a specific alarm trigger.

#### Retaining alarm status

The status of alarms is retained when the application is shut down. When the application is restarted:

- if an alarm is still active, the time begins accumulating again.
- if an alarm is still active, the number of times the alarm has been triggered is increased by 1.

#### Changing the alarm status displayed in the alarm status list

When the operator presses the alarm status mode button, the type of alarms displayed in the alarm status list changes as follows:

- If the current type displayed is all alarms, the list changes to display active alarms only.

- If the current type displayed is active alarms, the list changes to display past alarms only.
- If the current type displayed is past alarms, the list changes to display all alarms.

The change affects the linked or selected alarm status list only, not all alarm status lists.





# 10 Setting up FactoryTalk Diagnostics

This chapter describes:

- FactoryTalk® Diagnostics.
- displaying diagnostics messages during application development.
- viewing FactoryTalk Diagnostics log files.
- using the Diagnostics Setup tool.
- displaying and printing diagnostics messages at run time.
- the [DIAGNOSTICS] display.
- creating your own diagnostics display.
- opening and closing the diagnostics display.
- how the diagnostics list graphic object works.

## About FactoryTalk Diagnostics

FactoryTalk Diagnostics records information about various types of system activity, including:

- macro usage.
- operator comments.
- system messages and errors.
- errors from the communication network.
- tag read and write activity.

## Browsing diagnostics messages

You can view diagnostics messages about system activity during application development and at run time. For example, while you develop your application, use diagnostics messages to track what the system is doing and to ensure you've set things up the way you intend. At run time, use diagnostics messages to show the operator messages about system activity. You can also save information in a log file for future processing or analysis.

## How to set up FactoryTalk Diagnostics

You must set up FactoryTalk Diagnostics on both the development computer and the runtime computer. The settings for FactoryTalk Diagnostics apply to all Rockwell Automation® products installed on the computer.

Setting up FactoryTalk Diagnostics involves:

- setting up destinations, where logged information is received for storage or display.
- setting up which destinations receive which categories of messages. This is called message routing.
- setting up how messages are displayed and printed at run time.

This chapter describes how to set up FactoryTalk Diagnostics on personal computers. For information about setting up FactoryTalk Diagnostics on the PanelView™ Plus or PanelView™ Plus CE terminal, see the *PanelView Plus Terminals User Manual*. This manual is available on the FactoryTalk® View Machine Edition CD.

## Destinations

FactoryTalk Diagnostics allows you to send diagnostics messages to multiple destinations.

### During application development

During application development, you can send diagnostics messages to:

- the Diagnostics List at the bottom of the FactoryTalk View Studio window.
- a local log, which is viewable using the FactoryTalk Diagnostics Viewer tool.
- an ODBC database.

### At run time

At run time, you can send diagnostics messages to:

- diagnostics list graphic objects, either in the default [DIAGNOSTICS] display, or in a display you create.
- a printer
- a local log (available on personal computers only), which is viewable using the FactoryTalk Diagnostics Viewer.
- a remote log (PanelView Plus or PanelView Plus CE terminals only).
- an ODBC database.

The FactoryTalk Diagnostics Viewer is available on personal computers only. However, if you send diagnostics messages from a PanelView Plus or PanelView Plus CE terminal to a personal computer, you can view the messages in the FactoryTalk Diagnostics Viewer.



The destinations available on your computer might vary, depending on which Rockwell Automation products you have installed.

## Message routing

You can decide which destinations receive which kinds of messages. This allows you to make the right information available to the right person, in the right place.

For example:

- You can send messages that contain information about what the system is doing to the local log file. You can also send warnings about things that might go wrong if left unattended to the local log file.

This allows a control systems engineer to analyze system activity and performance, and make corrections during scheduled maintenance times.

- You can send errors that require immediate action to the diagnostics list object in the [DIAGNOSTICS] display, as well as the log file.
- You can set up the [DIAGNOSTICS] display to open automatically when system activity occurs, to allow the operator to respond to problems that might stop production if they are not solved immediately.

## Categories

FactoryTalk Diagnostics categorizes messages by severity, and by audience. For each destination, you can specify the message severity and audience.

Message categories		Error	Warning	Info	Audit
		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Operator		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Engineer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Developer		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Secure		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## Choosing not to log messages



If messages of a particular category are not routed to any destination, the messages for that category are not logged. For example, you might not want to log information messages, or you might not want to send any messages to the Operator.

## Message severities

FactoryTalk View categorizes messages in four severities:

- Errors indicate that a process or action has failed. For example, a tag's value could not be written, or a user typed the wrong password while logging on to the system.
- Warnings indicate that a process or action might not function correctly, or might eventually fail if preventive action isn't taken. For example, if an ActiveX® control

used in a graphic display is a different version than the one installed on the runtime computer, a warning is logged to indicate the mismatch. Mismatched ActiveX controls might not behave as expected at run time.

- Information indicates that a process or action has completed successfully. For example, a user logged on to the system or a tag value has been written to the data source.
- Audit indicates that the system configuration has been changed. FactoryTalk View records the creation, modification, and deletion of components (such as graphic displays) as audit messages.

Other Rockwell Automation products also use audit messages. For example, if you set up an audit log in RSMACC, FactoryTalk View's audit messages will be sent there, as well as to FactoryTalk Diagnostics.

In the FactoryTalk Diagnostics Viewer, the message severity is shown in the Severity column, indicated by the ! symbol.

The content of the messages is provided by the system—you don't need to do anything to set up message content.

## **Audiences**

FactoryTalk Diagnostics allows messages to be categorized differently for the various people for whom the messages are relevant. You can sort the messages in the FactoryTalk Diagnostics Viewer, so that those relevant to a particular person are grouped together.

### **Who receives which messages?**

You can specify which types of messages are sent to the Operator, Engineer, and Developer audiences.

FactoryTalk View automatically sends audit messages to the Engineer and Secure audiences. Audit messages allow auditing tools, such as those required for US Government 21CFR Part 11 compliance, to track system activity.

## **Displaying diagnostics messages during application development**

During application development, diagnostics messages are displayed as you create, modify, and delete components, and when you test your graphic displays. The messages are displayed in the Diagnostics List at the bottom of the FactoryTalk View Studio window. You can also view the messages in the FactoryTalk Diagnostics Viewer, as described in the next section.



By default, all warning and error messages are displayed in the Diagnostics List. To change what is displayed in the Diagnostics List, use the Diagnostics Setup tool, as described on page 10-6.

For information about testing graphic displays, see page 19-10.

### **To display the Diagnostics List**

1. On the View menu, click Diagnostics List. A check mark beside the menu item indicates the Diagnostics List is displayed.

For information about moving, resizing, and clearing messages in the Diagnostics List, see page 2-5.

### **If you don't want to display diagnostics messages**

1. On the View menu, click Diagnostics List. The option is turned off when no check mark appears.

## **Viewing FactoryTalk Diagnostics log files**

Use the FactoryTalk Diagnostics Viewer tool to view the contents of FactoryTalk Diagnostics local log files. You can open the FactoryTalk Diagnostics Viewer from within FactoryTalk View Studio, or from the Windows® Start menu.

The FactoryTalk Diagnostics Viewer is available on personal computers.

You can also set up personal computers to accept messages from PanelView Plus or PanelView Plus CE terminals, and then display the messages in the FactoryTalk Diagnostics Viewer or in the Diagnostics List. For information about sending messages from PanelView Plus or PanelView Plus CE terminals to personal computers, see page 10-8.

### **To open the FactoryTalk Diagnostics Viewer, do one of the following**

- In FactoryTalk View Studio, on the Tools menu, click Diagnostics Viewer.
- On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk Tools, and then click Diagnostics Viewer.

For information about setting up FactoryTalk Diagnostics to send messages to the local log, see Help for the Diagnostics Setup tool. For additional information about setting up and using the FactoryTalk Diagnostics Viewer, see the FactoryTalk Diagnostics Viewer Help.

## Using the Diagnostics Setup tool

Use the Diagnostics Setup tool to set up:

- destinations and message routing on the development computer, including logging to the local log or to an ODBC database.
- destinations and message routing on personal computers, including runtime logging to the local log or to an ODBC database.
- the Windows CE runtime destination to receive logged messages from.

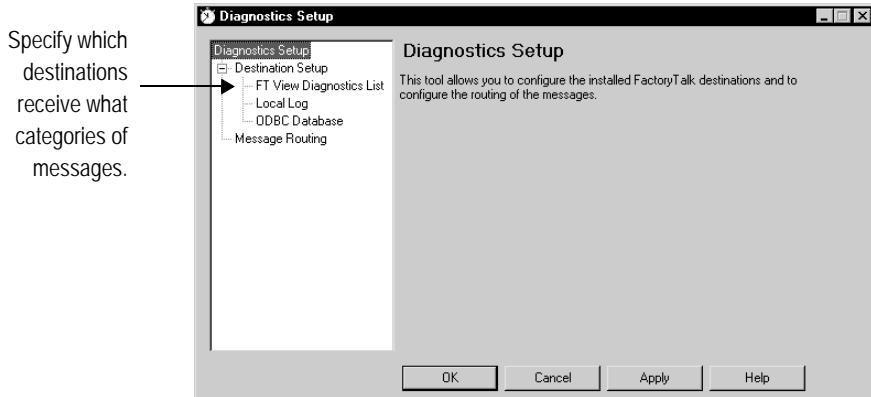
You can also use the Diagnostics Setup tool to clear the local log file.

### To open the FactoryTalk Diagnostics Setup tool, do one of the following

- In FactoryTalk View Studio, on the Tools menu, click Diagnostics Setup.
- In the FactoryTalk View ME Station dialog box, click Terminal Settings, and then double-click Diagnostics Setup.

For information about opening the FactoryTalk View ME Station dialog box:

- on personal computers, see page 15-4.
- on a PanelView Plus or PanelView Plus CE terminal, see page 16-3.
- On the Windows Start menu, select Programs, Rockwell Software, Utilities, and then click Diagnostics Setup.



For details about using the Diagnostics Setup tool, see Help for the tool.

## Logging to an ODBC database

The option of logging FactoryTalk Diagnostics messages to an ODBC database is available for personal computers only.

ODBC logging works by periodically exporting the contents of the local log file to an ODBC-compliant database. FactoryTalk View supports these ODBC-compliant databases:

- Microsoft® Access
- Sybase SQL Server
- Oracle
- Microsoft SQL Server

If you have set up FactoryTalk Diagnostics to overwrite events in the local log file, make sure messages are logged to the ODBC-compliant database before the oldest events are deleted.

For information about the contents of the FactoryTalk Diagnostics ODBC tables, see Appendix D.

## Setting up message buffering

FactoryTalk Diagnostics messages are stored in the computer's local log file, and are exported to the ODBC-compliant database at the interval you specify.

You can also specify that messages remain in the local log file for a period of time after they have been exported to the ODBC-compliant database. This is useful in the event of a network failure, or any other reason that causes the database to be unavailable. In this case, the messages remain in the local log file until the buffer time expires. If the ODBC-compliant database becomes available during that time, the buffered messages are then exported to the database.

## Routing messages

For each destination you set up, you can specify how messages are routed.

### To route messages, specify:

- the types of messages to log.
- the audiences to send the messages to.
- whether to accept messages from Windows CE devices.

For information about who receives which types of messages, see page 10-4.

## Receiving messages from a PanelView Plus or PanelView Plus CE terminal

You can set up FactoryTalk Diagnostics to receive messages that are generated on the PanelView Plus or PanelView Plus CE terminal at run time. The terminal must be on the same network as the computer running FactoryTalk Diagnostics.

To receive messages:

- enable the personal computer to receive the messages. For details see Help for the Diagnostics Setup tool.
- set up the PanelView Plus or PanelView Plus CE terminal to send the messages.

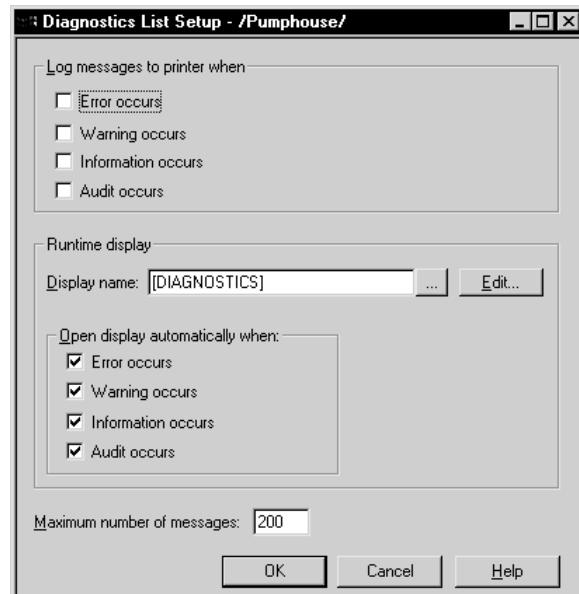
For information about setting up a PanelView Plus or PanelView Plus CE terminal to log to the personal computer, see the *PanelView Plus Terminals User Manual*.

## Displaying and printing diagnostics messages at run time

At run time, you can display diagnostics messages in a graphic display. You can also print the messages.

## Using the Diagnostics List Setup editor

Use the Diagnostics List Setup editor to set up when to print and display messages at run time in FactoryTalk View ME Station.



For details about using the Diagnostics List Setup editor, see Help.

Printed messages list the date and time, the category of system activity, and the message text. By default, messages are not printed at run time.

By default, your application is set up to display diagnostics messages automatically at run time. If you want to use the default settings, you don't need to do anything further to set up diagnostics messages for print and display in FactoryTalk View ME Station.

The messages that are printed and displayed depend on how you set up FactoryTalk Diagnostics on the runtime computer:

- For applications running on personal computers, use the Diagnostics Setup tool to set up diagnostics message logging. For information about using the Diagnostics Setup tool, see Help.
- For information about setting up FactoryTalk Diagnostics on the PanelView Plus or PanelView Plus CE terminal, see the *PanelView Plus Terminals User Manual*.

For applications running on personal computers, you can also view the runtime messages in the FactoryTalk Diagnostics Viewer, as described on page 10-5.

## If you don't want to display diagnostics messages in FactoryTalk View ME Station

1. In FactoryTalk View Studio, in the Diagnostics List Setup editor, clear the Display name box, or clear the four “Open display automatically when” boxes.

## Setting up how messages are displayed and printed at run time

Follow these steps to set up how diagnostics messages are displayed at run time:

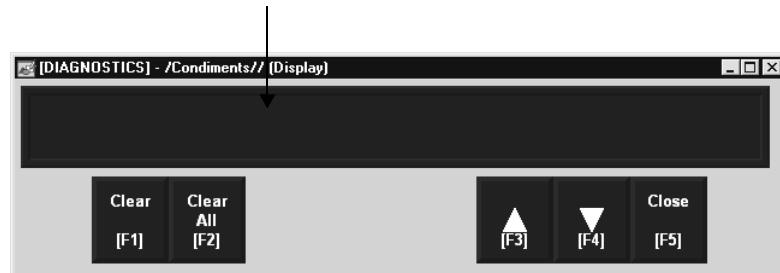
1. In FactoryTalk View Studio, in the Diagnostics List Setup editor, specify which messages to print, the graphic display to use, when to open the graphic display, and the maximum number of messages to store.
2. If desired, modify the default [DIAGNOSTICS] display, or create your own graphic display to use for diagnostics messages. For example, if you don't want the operator to clear all messages at once, edit the default display to remove the clear all button.

For information about graphic displays, see Chapter 19.

## The [DIAGNOSTICS] display

When you create an application, it comes with a graphic display called [DIAGNOSTICS]. The [DIAGNOSTICS] display is the default display for showing diagnostics messages at run time.

Diagnostics list graphic object.



You can use the [DIAGNOSTICS] display as is, or modify the display. For example, you can change the color of the objects, or add and remove buttons. Or, you can create your own graphic display to use for displaying diagnostics messages. In the Diagnostics List Setup editor, specify the display to use.

## The diagnostics list graphic object

The [DIAGNOSTICS] display contains a diagnostics list graphic object, which lists the diagnostics messages. For more information about the diagnostics list, see page 10-11.



## Buttons in the [DIAGNOSTICS] display

The [DIAGNOSTICS] display contains diagnostics list buttons for clearing the messages, as well as buttons for scrolling the list and closing the display. For information about how the buttons work, see page 10-12.

## Creating your own diagnostics display

You can create your own graphic display for displaying diagnostics messages, containing a diagnostics list graphic object and the buttons the operator needs for using the display.

If you create your own graphic display, use an On Top display and select the Cannot Be Replaced option.

For more information about the diagnostics list, see page 10-11. For information about the buttons you can use with the list, see page 10-12. For information about creating graphic displays and graphic objects, see Chapter 19 and Chapter 20.

## Opening and closing the diagnostics display

### Opening the display

The diagnostics display you specify in the Diagnostics List Setup editor (either the default [DIAGNOSTICS] display or a display you create) is automatically opened whenever a diagnostics message of the specified severity occurs.

The display remains open until the operator closes the display.

If you decide not to open the diagnostics display automatically, you can create a goto display button that the operator can press to open the diagnostics display. For information about setting up a goto display button and specifying the display to open, see Help.

### Closing the display

The operator can close the display by pressing a close display button.

## How the diagnostics list graphic object works

At run time, when a severity of system activity occurs that you've set up to log, a message about the activity is added to the diagnostics list. The diagnostics list can be in the [DIAGNOSTICS] display, in a diagnostics display you have created, or can be placed on any display in your application.

You can use multiple diagnostics lists, in the same display or in different displays. Each diagnostics list displays the same information.

For information about setting up diagnostics lists, see Help.

## What is displayed

- If the message is too long to fit in the list, the last displayed character is replaced with an asterisk (\*).
- The number of diagnostics messages that is visible at one time depends on the height of the diagnostics list and the list's font size.
- Messages are listed with the newest diagnostics message on top.

## Using buttons with the diagnostics list

### Diagnostics buttons

You can use these diagnostics buttons to interact with the diagnostics list:

This button	Does this
Diagnostics clear	Clears the selected message from all diagnostics lists in the application.
Diagnostics clear all	Clears all diagnostics messages from all diagnostics lists in the application.

You can assign any captions you choose to the labels on the diagnostics buttons.

You can link the diagnostics clear button to a specific diagnostics list. When the operator presses the button, the selected message is deleted from the linked diagnostics list, and also from any other diagnostics lists in the application. For information about linking buttons to objects, see page 21-9.

### Key buttons

You can also use these key button graphic objects to interact with the diagnostics list:

This button	Does this
Move up	Scrolls up one row in the list.
Move down	Scrolls down one row in the list.
Page up	Moves the highlight up one page in the list.
Page down	Moves the highlight down one page in the list.
Home	Moves the highlight to the top message in the list.
End	Moves the highlight to the bottom message in the list.



You can set up the key buttons to work only with the specified diagnostics list, or to work with whichever object is selected in the graphic display. For information about linking buttons to objects, see page 21-9.

For information about creating graphic objects, see Chapter 20. For information about setting up specific buttons, see Help.



This chapter describes:

- using FactoryTalk Security™.
- steps for setting up security without FactoryTalk Security.
- using the Runtime Security editor.
- how user accounts and security codes work.
- assigning security to graphic displays.
- providing a way for users to log in and log out.
- preventing unauthorized users from stopping the application.
- setting up FactoryTalk Security in FactoryTalk® View Studio, including:
  - adding FactoryTalk Security user groups and users, and setting up passwords.
  - assigning user access to the computer's FactoryTalk Directory. FactoryTalk View Machine Edition uses the local FactoryTalk Directory.
  - setting up security access to the application, action groups, policies, networks and devices, user groups, users, and other resources.
  - setting up policies for passwords and user accounts, for backing up and restoring FactoryTalk Directory, and for tracking activities for audit purposes.
- a sample startup display that uses all the FactoryTalk View security features.

## Using security with your application

FactoryTalk Security works together with FactoryTalk View security features to provide secure access to applications during development and at run time. FactoryTalk Security is installed as part of the FactoryTalk Services Platform, and uses FactoryTalk Directory to manage user accounts and policies.

FactoryTalk Security applies to all applications and all Rockwell Automation® products installed on the development or runtime computer. For example, when you add a FactoryTalk Security user, the user is then available to add to any FactoryTalk View application.

You can use FactoryTalk Security features on the development computer to:

- determine which users can view, edit, create, and delete applications.
- determine which users can manage users and set up security.

- determine which users can back up and restore applications.

You can use a combination of FactoryTalk Security and FactoryTalk View security features at run time to:

- prevent unauthorized users from running the application.
- prevent users from opening graphic displays.
- prevent unauthorized users from stopping the application.
- prevent users from seeing and using certain parts of a graphic display.
- prevent users from switching to other applications. This feature uses the DeskLock tool, and is available for personal computers only. You can use this tool to set up a customized desktop and to prevent access to Windows® functions. For more information about the DeskLock tool, see Chapter 15. The DeskLock tool does not use FactoryTalk Security.

To limit the users who have access to the application or parts of the application, you must set up user accounts and passwords.



If you set up the data source to open graphic displays remotely (using global connections), remote display changes occur whether or not the logged-in user has security access to a given display.

For details about controlling display changes remotely, see page 8-4.

## If you don't want to use FactoryTalk Security

If you don't plan to use security for your application, you can eliminate or minimize the use of FactoryTalk Security.

During installation, users are set up as follows:

- an administrative user named Administrator is created.
- all users are granted initial access permissions.
- a Windows-linked user group named Authenticated Users is created and added to the User Groups folder in the FactoryTalk Directory.

This means that any user who is authenticated by Windows—that is, has a password that allows access to the development computer—can open FactoryTalk View Studio without providing a user name or password.



Authenticated users are not supported on Windows CE. On Windows CE terminals, Windows-linked users must be added to FactoryTalk Directory before they can be logged in.

## If you don't use FactoryTalk View user accounts

FactoryTalk View creates a user named DEFAULT, and this user is automatically logged in when you start applications at run time. You don't need to set up any other users, and users don't need to log in to use the applications you create.

## If you use FactoryTalk View user accounts

If you want to use FactoryTalk View user accounts, for example, to run login and logout macros, but you don't want to use FactoryTalk Security, follow these steps.

1. Add users or user groups to FactoryTalk Security. For details, see page 11-17.
2. Add the FactoryTalk Security users or user groups to the FactoryTalk View Runtime Security editor. For details, see page 11-4.
3. In the FactoryTalk View Runtime Security editor, if desired, assign login and logout macros and security codes to users or user groups. For details, see Help for the editor.
4. In the Graphics editor, assign security codes to graphic displays that will have limited access.
5. Provide users with a method for logging in and out.
6. If desired, prevent unauthorized users from stopping the application.

These steps are described in detail in this chapter.

## Working with the Runtime Security editor

Use the Runtime Security editor to:

- add FactoryTalk Security users and assign security codes and login and logout macros, for each user or group of users that will have access to the application at run time.
- migrate users from RSView® 3.20 and earlier to FactoryTalk View version 5.00.
- assign user passwords (version 3.20 and earlier applications only).
- add Windows users or groups (version 3.20 and earlier applications only).

The method for adding users and assigning security codes to them depends on whether you are setting up users for FactoryTalk View 5.00 or RSView 4.00 applications or for RSView 3.20 and earlier applications. There is a separate tab for 4.00 and later applications and for 3.20 and earlier applications.

The Runtime Security editor has special items on the View menu to control the appearance of the editor. It also has items on the Setup menu, and extra tools on the toolbar.

## The DEFAULT user

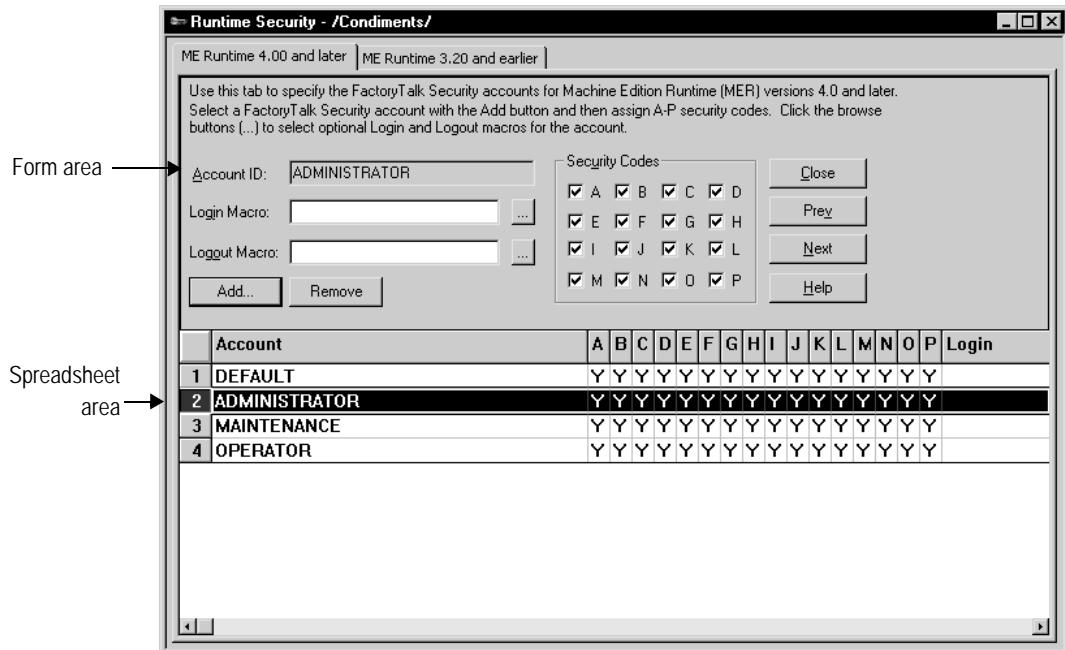
The first account in the editor is the DEFAULT user account. The DEFAULT account is used when no one is logged in. The DEFAULT user initially has access to each security code. Unless you want everyone to have access to all parts of the application at run time without logging in, turn off the DEFAULT user's access to the security codes you plan to use.

You cannot delete the DEFAULT user account.



Assign the security code for the startup display to the DEFAULT user, or else the startup display won't open. If the startup display uses the \* security code, you can assign any code from A to P to open the display. For more information, see page 11-9.

## Setting up users for 4.00 and later applications



Use the form area to assign macros and security codes to users.

### To add FactoryTalk Security users and groups to FactoryTalk View and assign security codes to them

1. Open the Runtime Security editor and select the ME Runtime 4.00 and later tab.

2. Click Add.



3. In the Select User or Group dialog box, add users and groups, and select the ones to add to FactoryTalk View. You can only select one user or group at a time.

For details on using the Select User or Group dialog box, see Help.

4. In the Runtime Security editor, assign login and logout macros and security codes to the users and groups.

For details about using the Runtime Security editor, see Help.

### To migrate RSView 3.20 and earlier users to FactoryTalk View 5.00

1. With the Runtime Security editor open and the ME Runtime 4.00 and later tab selected, on the Setup menu, click Migrate ME Runtime 3.20 and earlier accounts.

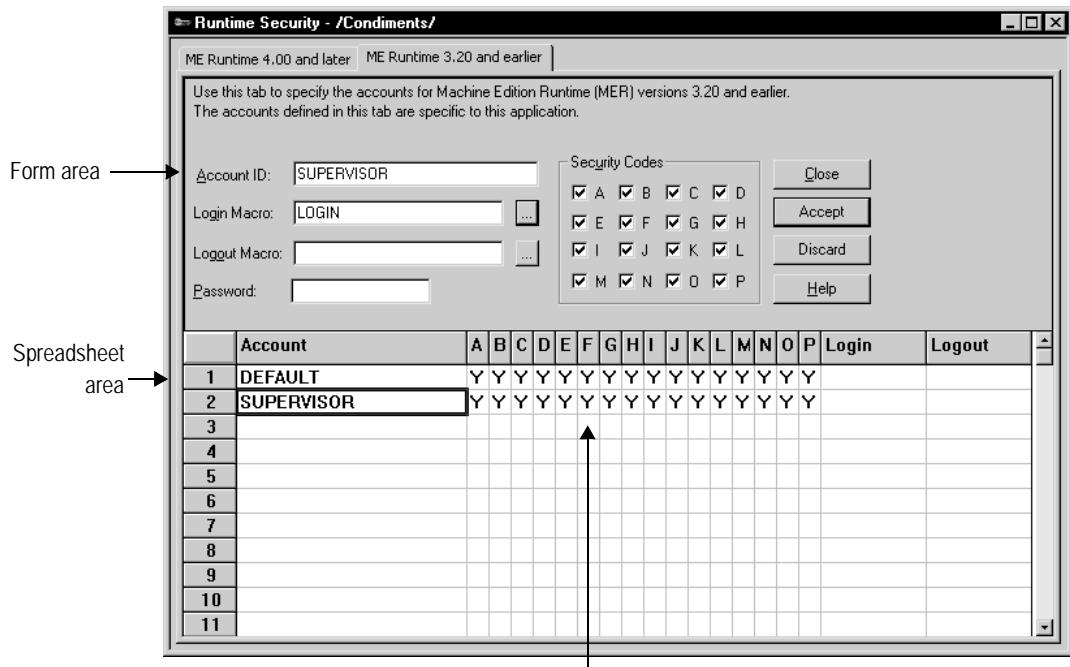
Users from the ME Runtime 3.20 and earlier tab are added to the FactoryTalk Security Users folder and to the ME Runtime 4.00 and later tab.



Users are not migrated if they already exist in the ME Runtime 4.00 and later account list.

2. If desired, edit the users' security codes and login and logout macros.
3. To edit the users' passwords, see page 11-18.

## Setting up users for 3.20 and earlier applications



### To add users to RSView 3.20 and earlier applications and assign security codes to them

1. Open the Runtime Security editor and click the RSView 3.20 and earlier tab.
2. Add users, and then assign macros, passwords, and security codes to them.



For details about using the Runtime Security editor, see Help.

### **Changing RSView 3.20 and earlier user passwords**

For RSView 3.20 and earlier applications, users cannot change passwords at run time. You can only change passwords in the Runtime Security editor. After changing passwords, recreate the application and download the .mer file.

### **Adding 3.20 and earlier users or groups from a Windows domain**

In RSView 3.20 and earlier applications, there are two possible types of user accounts:

- users or groups from a Windows domain
- users you create in FactoryTalk View.

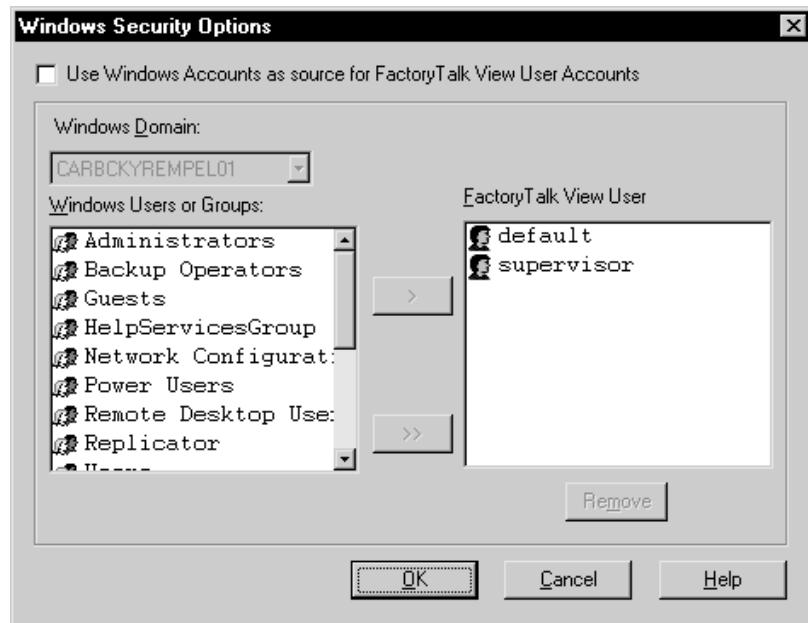
You cannot use both user account types in the application at the same time.

Before you can make use of Windows user accounts in FactoryTalk View, you must first create the user accounts in your Windows domain. To do this in a Windows network domain, your network must have at least one Windows server acting as a domain controller. You cannot use Windows Workgroups.

When you use Windows users or groups, the user password is the Windows domain password.

#### **To add users from a Windows domain to FactoryTalk View**

1. Open the Runtime Security editor, and then click the RSView 3.20 and earlier tab.
2. On the Setup menu, click Windows Security Options.



3. In the Windows Security Options dialog box, specify the users and groups to use. For details, see Help.
4. In the Runtime Security editor, specify login and logout macros and assign security codes to the users. For details, see Help.

You cannot change the user or group name (Account ID) and password. To change the password, use the Windows Control Panel.

### Removing 3.20 and earlier users or groups

You can also use the Windows Security Options dialog box to remove users or groups. When you remove a user or group from the FactoryTalk View User list, the user or group's name is deleted from the FactoryTalk View Runtime Security editor. The user or group is not deleted from the Windows domain.

If you remove users from the Windows domain, they are not removed automatically from the FactoryTalk View Runtime Security editor. You must remove the users from the Runtime Security editor manually, using the Windows Security Options dialog box.



## How user accounts and security codes work

When you assign a security code to a graphic display, only the users and groups who are assigned that security code have access to the display. There are 16 security codes, A through P.

The asterisk symbol (\*) is used in displays and represents any code. Users must have at least one code assigned to open displays that use the \* code.

You can assign combinations of security codes to users and groups, to allow each user or group access to a different set of displays or graphic objects.

To use security codes to restrict user access to graphic objects, assign visibility animation to the objects using expressions containing the security functions. For an example, see page 11-11.

## Assigning security to graphic displays

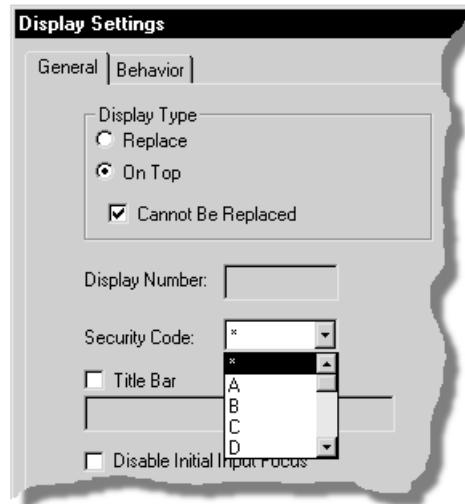
Assign security to graphic displays in the Graphics editor. You can:

- assign security when you create the graphic display, or at any time after creating the display.
- assign the same security code to more than one display.
- assign whichever codes you want, in any order. For example, you can choose to use only the codes D and P and you can assign P before you assign D.

For more information about graphic displays, see Chapter 19.

### To assign security to a graphic display

1. In the Graphics editor, right-click an empty area of the display, and then click **Display Settings**.



2. In the General tab, select a security code.

To give every security code access to the display, select the asterisk (\*). The \* is selected by default.

---

#### **Example: Assigning security codes to prevent access to graphic displays**

This example shows how to set up users with access to different graphic displays.

1. Assign security codes to graphic displays as follows:

To this display	Assign this security code
Alarm History	A
Boiler	B
Furnace	C
Top Secret	D

2. In the Runtime Security editor, assign the security code P to the DEFAULT user.

3. Assign security codes for the displays they can use to the other users:

To this user	Assign these security codes
MAINTENANCE	A, B, C
OPERATOR	A, B
ADMIN	All

The users have access to these graphic displays:

- MAINTENANCE has access to the first three graphic displays.
- OPERATOR has access to the Alarm History and Boiler displays, but not to the Furnace or Top Secret displays.
- The ADMIN user has access to every graphic display.
- The DEFAULT user has access only to graphic displays to which \* or P has been assigned. In this example, the DEFAULT user does not have access to any of the four graphic displays.

In this example, you could have assigned the same security code to the Alarm History and Boiler displays, since all users except the DEFAULT user have access to these displays.

---

---

### **Example: Using security codes to control the visibility of the shutdown button**

This example shows how to set up users so that only authorized users can use the shutdown button to stop the application.

This example uses the CurrentUserHasCode security function to determine whether a user is authorized to view and use the shutdown button.

The security code E is assigned to users who are authorized to use the button.

1. In the Runtime Security editor, clear all the security codes for the DEFAULT user.

2. Assign security codes to the other users:

To this user	Assign these security codes
OPERATOR	A, B
MAINTENANCE	A, B, C, E
ADMIN	All

3. Create a shutdown button.
4. Assign visibility animation to the button using this expression:

`CurrentUserHasCode( E )`

For the Expression True State, click Visible.

When the graphic display containing the shutdown button is open, the MAINTENANCE and ADMIN user can see the button. If the OPERATOR or DEFAULT user is logged in, the button is not visible.

---

## Providing a way for users to log in and log out

Use login and logout buttons to provide a way for users to log in and log out. Make sure you place these buttons in graphic displays that all users have access to.

### Logging in

When the application starts, the DEFAULT user is logged in. If a macro is assigned to the DEFAULT user, the macro runs.

For example, you could create a macro to set the remote display number to open a new login display after the DEFAULT user logs in. For information about controlling display changes remotely, see page 8-4. Remote display changes are not checked for security.

### To provide a way for users to log in

1. Specify the startup display in the Startup editor, in the Initial graphic box. The startup display is the graphic display that opens when your application starts running.

The startup display must have the \* code or a security code that is assigned to the DEFAULT user, or it will not open.

For information about the Startup editor, see Chapter 14.

2. Create a login button in the startup display.

For information about creating graphic objects, see Chapter 20. For information about setting up login buttons, see Help.

For more information about logging in, see Chapter 17.

## **Logging out**

To best maintain your security system, set up your application so that users who log in are also required to log out.

When the current user logs out, if a logout macro is assigned to the user, the macro runs. If the user belongs to a group, and a logout macro is set up for the group, the group's logout macro runs.

After the current user is logged out, the DEFAULT user is logged in. If a login macro is assigned to the DEFAULT user, the macro runs. Turn off the DEFAULT user's access to the security codes you plan to use.

### **To provide a way for users to log out**

1. Create a logout button in a graphic display that all logged-in users have access to.

## **Logging out automatically**

You can set up your runtime terminal so that the current user will be logged out automatically after the terminal has been inactive for a specified period of time. The automatic logout is independent of FactoryTalk Security settings or Windows security settings. The logout will take place regardless of what you set up in FactoryTalk Security or Windows.

### **To set up automatic logout**

1. In the Explorer window, double-click Project Settings, and then click the Runtime tab.
2. Select Enable auto logout.
3. Specify how many minutes the terminal must be inactive before the automatic logout occurs.
4. If desired, specify a graphic display to open after the automatic logout.

At run time, when the inactivity period expires, the current user is logged out. Logout macros assigned to the current user run. The DEFAULT user is logged in. The DEFAULT user's login macro runs. The specified graphic display opens (if any).

## Preventing unauthorized users from stopping the application

### To prevent unauthorized users from stopping the application

- Do not use a title bar on the application window, because the title bar contains a Close button. To turn off the title bar for the application window, in the Project Settings editor, in the Runtime tab, clear the Title bar check box.

For more information about using the Project Settings editor, see Chapter 4.

- Assign visibility animation to the shutdown button, so that the button is only visible when a user who is authorized to stop the application is logged in. For an example, see page 11-11.

Or, set up security for the graphic display in which the shutdown button is located, so only users who are authorized to stop the application can open the display.

### Additional step for personal computers

- Use the DeskLock tool to prevent users from having access to the Windows desktop. This will prevent access to operating system functions such as restarting Windows or shutting down tasks. For more information, see page 15-17.

### Additional step for PanelView Plus CE applications

- Hide the Windows Taskbar to make it inaccessible.

### To hide the Windows Taskbar

1. On the Windows Start menu select Settings, and then select Taskbar.
2. Clear the “Always on top” check box.
3. Select Auto hide.
4. When you are finished, click OK.

5. At the Command prompt or in the Run box, run Regflush.exe to save your changes to persistent internal flash memory. Otherwise, the changes will be lost when you reboot.

## Setting up FactoryTalk Security for your application

In the Explorer window of FactoryTalk View Studio you can:

- create FactoryTalk Security users and groups of users, and assign passwords to users.
- set up access to the FactoryTalk Directory.
- set up access to the application.
- assign security access to action groups, policies, groups, and users.

- set up access to network devices, connections, and databases.
- specify which users can backup and restore FactoryTalk Directory.
- specify the types of activities to track for audit purposes.
- specify policies for passwords and user accounts, such as how often passwords must be changed.

Depending on what other products are installed on the computer, you can also assign security access for those products to your users. For example, you can specify which users are allowed to browse the network for RSLinx® Enterprise™ devices.

Set up access to the FactoryTalk

Directory.

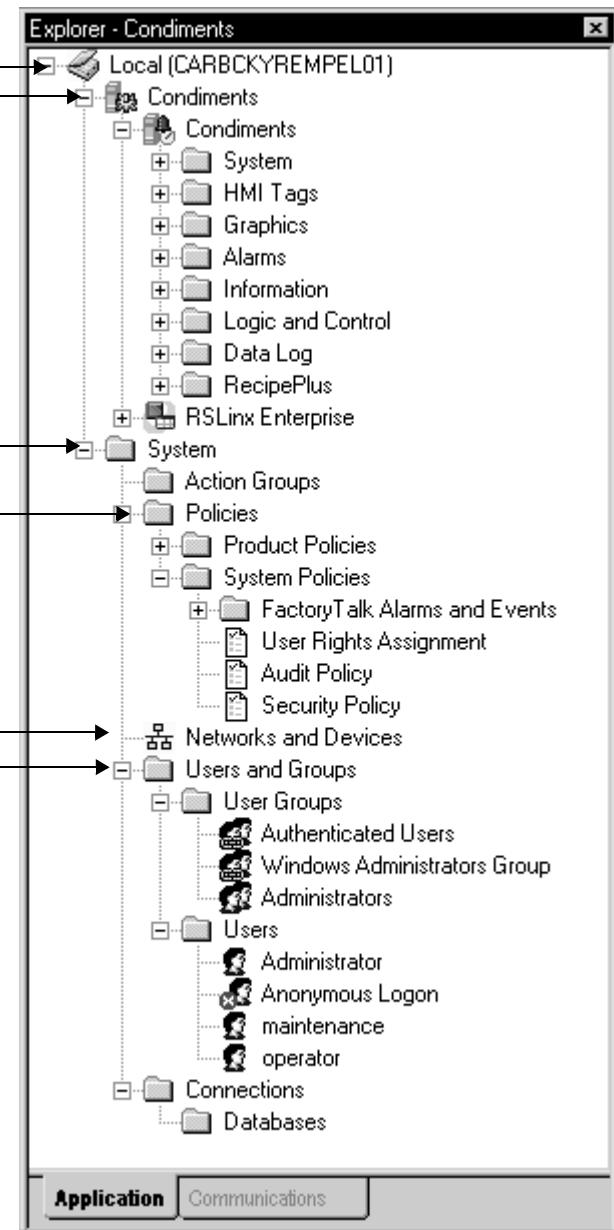
Set up access to the application.

You can assign security to all the  
folders in the System folder.

Set up policies.

Set up access to devices.

Add users and groups, and  
assign passwords. The link  
symbol indicates Windows-linked  
groups and users.



## Summary of steps

Follow these steps to set up security using FactoryTalk Security and FactoryTalk View security features:

1. In the Explorer window, in the System folder underneath the HMI server, add FactoryTalk Security user groups and users, and set up their passwords.

You can also add users and groups in the Runtime Security editor, in the ME Runtime 4.00 and later tab, as described on page 11-4.
2. Assign user access to the computer's FactoryTalk Directory.
3. If desired, restrict access to the application and to networks and devices. You can also restrict security access to policies, user groups, and users.
4. Set up policies for:
  - backing up and restoring FactoryTalk Directory.
  - tracking activities for audit purposes.
  - passwords and user accounts.
5. In the Runtime Security editor, add FactoryTalk Security users or user groups to FactoryTalk View, and assign login and logout macros and security codes to them.
6. In the Graphics editor, assign security codes to graphic displays that will have limited access.
7. Provide users with a method for logging in and out.
8. If desired, prevent unauthorized users from stopping the application.

These steps are described in detail in the following sections.

You can also assign security to action groups, connections, and databases. These topics are not discussed in this chapter. For information about these topics, see the FactoryTalk Security help.

## Creating FactoryTalk Security users

You can set up users in FactoryTalk Security before you add them to the FactoryTalk View Runtime Security editor.

FactoryTalk Security allows you to reference user accounts that have already been set up in Windows. These are called Windows-linked users. The link symbol in the Explorer window indicates that a user is a Windows-linked user.

The FactoryTalk Security access you give to users determines what actions they can perform in FactoryTalk View, both on the development computer and at run time.

To do this	Allow these Common actions
Open applications	Read and List Children
Add and edit applications or components	Write and Create Children
Delete applications or components	Delete
Assign security and create users	Create Children and Configure Security

You can assign different permissions to users at different levels of the application.

### Inheriting permissions

The security you set up at the top level, the FactoryTalk Directory, is inherited by the application and users below it, unless you select “Do not inherit permissions.” One way to set up security is to assign the Read and List Children permissions to the All Users group at the FactoryTalk Directory level, and then assign permissions for writing, creating children, configuring security, and deleting to users and groups at the application level.



If you deny permission for an action to a user, this overrides permission the user is granted as a member of a group. If you deny permission for an action to a group, allowing the action to a member of the group does not override the denial. The denial always takes precedence.

### To set up users

1. In the System folder, right-click Users, and then click New User or New Windows-Linked User.
2. Specify the user name, password, and password options. For details, see Help.
3. Right-click Users, and then click Security.
4. Specify what actions the users can perform. For details, see Help.
5. To add users to a group, right-click the group and then click Properties. For details, see Help.

### To change user passwords

1. In the Users folder, right-click the user name, and then click Properties.
2. Click Reset Password. For details, see Help.

Users can change their password at run time using the Password button. For more information, see page 17-4.

## Creating FactoryTalk Security user groups

The use of user groups is optional but recommended. Groups allow you to set up security access once for the entire group, rather than individually for members of the group. You can still set up separate permissions for individual members of the group.



Setting up groups saves time. You can add all the users of an application to a single group in FactoryTalk Security, then add the group to the FactoryTalk View Runtime Security editor in one step, which is much quicker than adding the users one by one.

FactoryTalk Security allows you to use groups that have already been set up in Windows. These are called Windows-linked groups. The link symbol in the Explorer window indicates that a group is a Windows-linked group.

FactoryTalk Security comes with an Administrators group already set up. To start with, members of this group have full security access to FactoryTalk Security, and therefore have full access to your applications. The user you set up for FactoryTalk Local Directory when you installed FactoryTalk Services Platform is a member of the Administrators group.



To ensure that you are never locked out of FactoryTalk Directory we recommend that you assign more than one user to the Administrators group. That way if one user is locked out, you can use the other one to log in.

### To set up user groups

1. In the System folder, right-click User Groups, and then click New User Group or New Windows-Linked User Group.
2. Add user groups. You can add members to the groups now or later. For details, see Help.
3. Right-click User Groups, and then click Security.
4. Specify what actions members of the group can perform. For details, see Help.
5. To add members to the group later, right-click the group and then click Properties. For details, see Help.

### Setting up security access to the FactoryTalk Directory

The FactoryTalk Security access that you set up for the FactoryTalk Directory applies to all FactoryTalk View applications and all Rockwell Automation products installed on the development computer.

By default, applications and the System folder inherit the security permissions assigned to the FactoryTalk Directory. This means that you only need to assign permissions to user groups or users once. If desired, you can override the permissions that you set up at this

level by assigning different permissions to applications and to the items in the System folder.

You can assign permissions for actions related to FactoryTalk View applications and tags here or at the application level. If you assign them here, they are inherited by all applications.

### **To set security permissions for the FactoryTalk Directory**

1. At the top of the tree in the Explorer window, right-click Local, and then click Security.
2. Specify which users and groups have access to FactoryTalk Directory, and assign permissions to users and groups for the actions listed in the dialog box. For details, see Help.

### **Setting up security access to the application**

Assign FactoryTalk Security at the application level to specify which users or groups can open the application and add, edit, and delete components. You can also specify which users or groups can write to tags and set up, view, and respond to alarms.

### **To set security permissions for the application**

1. Right-click the application name, and then click Security.
2. Specify which users and groups have access to the application, and assign permissions to users and groups for the actions listed in the dialog box. For details, see Help.

### **Setting up security access to System policies, groups, and users**

You can assign security access to the System folder and to all the folders in the System folder, including policies, groups, and users. You do this by specifying which groups and users have access to the folder, and what actions they can perform.

### **To set up security access to policies, groups, and users**

1. Right-click the System folder, or a folder in the System folder, and then click Security.
2. Specify which groups and users have access to the folder, and what actions they can perform. For details, see Help.

### **Setting up security access to networks and devices**

You can assign security access to networks and devices in general, to the development computer, and to specific devices. Set up security for networks and devices if you are using RSLinx® Classic™ devices.

## To set up security access to networks and devices

1. In the System folder, right-click Networks and Devices, the development computer workstation, or a specific device, and then click Security.
2. Specify which groups and users have access to the item, and what actions they can perform. For details, see Help.

## Specifying which users can back up and restore FactoryTalk Directory and FactoryTalk View applications

You can allow users or groups of users to back up and restore the contents of the local FactoryTalk Directory, including applications, users, groups, and all security settings.

Users must have permission to back up and restore FactoryTalk Directory to use the Application Manager tool to back up and restore applications. For information about using the Application Manager, see page 4-10.

## To specify which users can back up and restore FactoryTalk Directory and FactoryTalk View applications

1. In the System Policies folder, double-click User Rights Assignment.
2. Set permissions for users. For details, see Help.

## Specifying activities to track for audit purposes

You can keep an audit trail of activities such as changing the setup of the FactoryTalk Directory and whether security access is denied or granted to users.

## To specify which activities to track for audit purposes

1. In the System Policies folder, double-click Audit Policy.
2. Set up audit policies. For details, see Help.

## Specifying policies for passwords, accounts, and FactoryTalk sign-on

You can use the FactoryTalk Directory Security Policy to specify how to manage passwords and user accounts, including how long passwords must be, how often they must be changed (if ever), and whether to keep a record of deleted user accounts.

 Remember that any changes that you make to system security policies will be included as part of the run-time application. It is therefore very strongly recommended that you do not alter the default settings in the Security Policy Properties dialog box unless absolutely necessary, and unless you are aware of the full implications of the changes. Always test the run-time application to make sure the results are what you intended.

## Single sign-on

You can also use the Security Policy to turn off FactoryTalk single sign-on. If you turn this feature off, users must log into FactoryTalk each time it is FactoryTalk View Studio restarted. When the feature is on, if you shut down FactoryTalk View Studio and restart it without restarting your computer, you remain logged in to FactoryTalk and are not prompted to log in again.

## To specify how to manage passwords, user accounts, and FactoryTalk sign-on

1. In the System Policies folder, double-click Security Policy.
2. Set up options for accounts, passwords, and single sign-on. For details, see Help.

## Summary of security features in a sample startup display

This section provides an example of how to use FactoryTalk Security and FactoryTalk View security features in a startup graphic display called Log In.

### To create a startup display that uses security

1. Set up FactoryTalk Security user accounts and passwords.
2. Assign Common\Read and Common\List Children security permissions to all users that are allowed to run the application, at the FactoryTalk Directory or application level.
3. Assign Tag\Write Value permission to all users that are allowed to write values to tags.
4. In the Runtime Security editor, assign at least one security code to each user who you want to log in, including the DEFAULT user.
5. In the Project Settings editor, clear the Title bar check box.
6. Create a graphic display called Main Menu. In the display create navigation buttons that give the operator access to the different displays in the application. Assign a security code to the display so that only authorized users can open it.
7. Create a graphic display called Log In containing these button objects:

This button	Does this
Login	Opens the Login dialog box.
Logout	Logs the current user out and logs in the DEFAULT user.
Password	Opens the Change Password dialog box so the user can specify a new password.

This button	Does this
Goto display	Opens the Main Menu graphic display.
Shutdown	Shuts down the application.

8. Set up the goto display button in the Log In display to open the Main Menu display. If desired, attach visibility animation to the goto display button, so that only logged-in users can see the button.

For more information, see the example on page 11-24.

9. Attach visibility animation to the shutdown button, so that only authorized users can see (and therefore use) the button.

For more information, see the example on page 11-24.

10. Specify captions for the buttons, and add explanatory text to the graphic display. If desired, attach visibility animation to the text as well.

11. In the Display Settings dialog box, assign the security code \* to the Log In display.

12. In the Startup editor, in the Initial graphic box, select the Log In display.

Until an authorized user is logged in, further access to the application is denied. In addition, only authorized users can stop the application.

Users who have access to the application can use the application and log out. Make sure you provide a way for users to navigate back to the Log In display to log out.



When a user logs out, the DEFAULT user is automatically logged in. Do not give the DEFAULT user access to the goto display button or shutdown button.

You can also use the Remote Display Number connection to open the startup Log In display when a user logs out. This might be useful if users can log out from various locations (graphic displays) within the application. If you use the auto logout feature, you can specify that the Log In display is opened when an auto logout occurs.

For more information about	See
Using the Project Settings editor	Chapter 4
Attaching animation to graphic objects	Chapter 22
Setting up display navigation	Chapter 13
Using the Startup editor	Chapter 14

For more information about	See
Logging in, logging out, and changing passwords	Chapter 17
Stopping applications	Chapter 15

---

### **Example: Assigning visibility animation to the goto display button**

This example shows how to assign visibility animation to the goto display button so that a user must log in before seeing the button.

This example uses the security function CurrentUserName( ). The function returns the string value of the Account ID (user name) for the user who is currently logged in.

The CurrentUserName( ) function is case sensitive. All RSView 3.20 and earlier user names use uppercase letters.

When the application starts, the DEFAULT user is logged in. The DEFAULT user is also logged in when a user logs out.

#### **To assign visibility animation to the goto display button**

1. Right-click the goto display button, select Animation, and then click Visibility.
2. In the Animation dialog box, in the Expression box, type this:  
`CurrentUserName( ) <> "DEFAULT"`
3. For the Expression True State, click Visible.
4. Click Apply.

---

---

### **Example: Assigning visibility animation to the shutdown button**

This example shows how to assign visibility animation to the shutdown button so that only the ADMIN user can stop the application.

#### **To assign visibility animation to the shutdown button**

1. Right-click the shutdown button, select Animation, and then click Visibility.
2. In the Animation dialog box, in the Expression box, type this:  
`CurrentUserName( ) == "ADMIN"`
3. For the Expression True State, click Visible.

---

4. Click Apply.

---

---

### **Example: Assigning visibility animation to the shutdown button**

This example shows how to assign visibility animation to the shutdown button so that only users assigned code G or code H can stop the application.

#### **To assign visibility animation to the shutdown button**

1. Right-click the shutdown button, select Animation, and then click Visibility.
2. In the Animation dialog box, in the Expression box, type this:  
`CurrentUserHasCode(GH)`
3. For the Expression True State, click Visible.
4. Click Apply.

---





# 12 Setting up language switching

This chapter describes:

- what language switching is.
- summary of steps for setting up language switching.
- setting up Windows® to support language switching.
- adding languages to the application.
- exporting application text strings for translation.
- translating application text.
- importing translated text files.
- setting up multiple language support for graphic libraries.

## About language switching

The FactoryTalk® View language switching feature provides the ability to set up multiple languages for an application and switch languages dynamically at run time. You specify an initial language for the runtime application when you create it, and select the languages that will be available at run time. You can use up to 40 languages per development application, and 20 languages per runtime application.

When the application runs, operators can change the language using a language switch button. Set up a different language switch button for each language.

With language switching you can:

- develop an application in one language, export the user-defined text strings for the application, and then import translated strings for up to 40 languages into the same application.
- export an application's text strings in multiple languages into a single Microsoft® Excel spreadsheet.
- import text strings in multiple languages from a Microsoft Excel spreadsheet into your application in a single operation.
- use the same application in different countries, allowing operators in each location to view the application in their own language.
- allow operators in multilingual countries to use the language of their choice.

- import application components developed in different countries into a single application that supports multiple languages.

## **The default language**

For applications that use multiple languages, you can specify one of the languages as the default language. Any undefined text strings in the other languages can be displayed in the default language.

The default language is also used for RSView Machine Edition 4.00 applications. When you open or import the application into FactoryTalk View 5.00, the application's alarm, information, and local messages are associated with the language that you select to be the default language. For example, if you open an RSView Machine Edition 4.00 application that contains local messages, and you select French for the default language, the local messages will be treated as if they are French when you export the application's strings for translation. That is, they will appear in the French language column. You can add other languages for these messages, as described in this chapter.

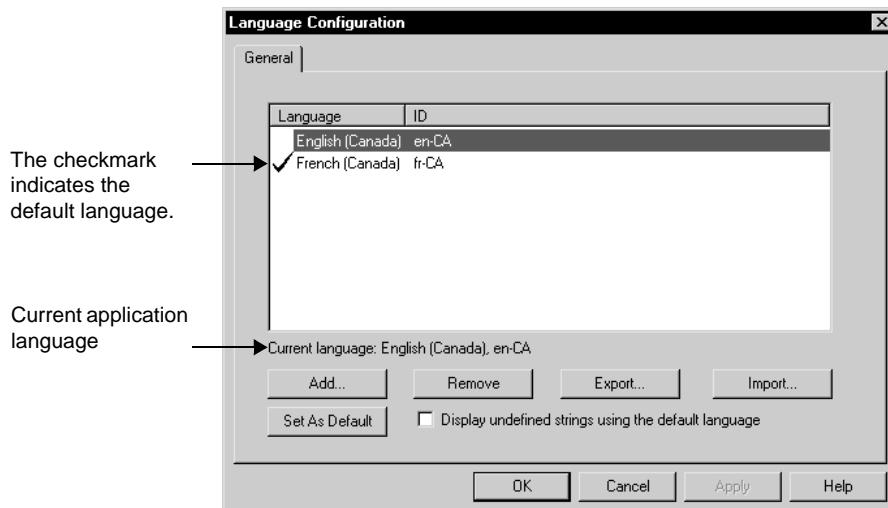
When you create a new application, the language you specify for the application is the default language. You can change the default language at any time. The default language can be any of the languages in the application.

You can display undefined strings in the default language during application development. If you do not select this option, undefined strings are displayed as a question mark “?” character. If you select this option, but do not include the default language with the runtime application, undefined strings are displayed as a question mark “?” character at run time.

Alarm trigger labels are displayed in the Trigger Label Selector list using the default language. If you change the default language, you must re-specify the trigger labels for every graphic object that filters alarms. To do this, specify the trigger label names in the new default language. For more information about filtering alarms when your application uses multiple languages, see page 9-5.

### **To change the default language**

1. On the Tools menu, click Languages.



2. Select the language to use for the default language.
3. Click Set As Default. A large checkmark to the left of the selected language indicates that it is now the default language.
4. To use this language for all undefined strings in the application, select Display undefined strings using the default language.

## Summary of steps

Follow these steps to set up language switching for an application:

1. For applications that will run on a personal computer, install the Windows languages that the application will use.
2. For applications that will run on a PanelView™ Plus or PanelView™ Plus CE terminal, set up the fonts that the application will use.
3. Create, open, or import the application in the language of your choice. For details, see Chapter 4.
4. Add languages to the application. For details, see page 12-5.
5. Create graphic objects and specify the text strings that they will use.
6. Create alarm, information, and local messages.
7. Create a language switch button for each language you plan to use at run time. For details, see Help.

8. Export the application text strings for translation. For details, see page 12-6.
9. Translate the text strings. For details, see page 12-9.
10. Import translated text strings for each of the translation languages. For details, see page 12-13.
11. Open the application in each language, to check the layout of the translated text.
12. Create the runtime application, specifying the startup language and the languages that the operator can switch to. For details, see Chapter 14.

## **Setting up Windows for language switching**

### **Installing Windows languages**

We recommend that you install all the languages that your application will use, on both the development and runtime computers. Installing languages turns on the Windows font support features, which allow applications to display characters for different languages using a single application font.

For details about installing languages, see Windows Help.

PanelView Plus and PanelView Plus CE terminals are shipped with languages already installed.

### **Setting up Windows fonts**

For applications that will run on a PanelView Plus or PanelView Plus CE terminal, install the fonts that the application will use. For applications that will run on a personal computer, the fonts you want are generally installed when you install the languages for the application.

For details about choosing fonts that work well with language switching, see page 20-13.

Make sure that the development and runtime computers are set up to use the fonts you want for the title bar and inactive title bar. Specify the font in the Windows Control Panel, in the Display Properties dialog box.

For information about setting up fonts on a PanelView Plus or PanelView Plus CE terminal, see the *PanelView Plus Terminals User Manual*. This manual is available on the FactoryTalk View Machine Edition CD.

### **Windows locale settings**

Windows locale formatting determines how the application displays time, date, and floating point values at design time and runtime.

When the application language is switched, the locale settings for the new language are used even if that language has not been installed.

You do not need to edit the default locale settings.

## Adding languages to the application

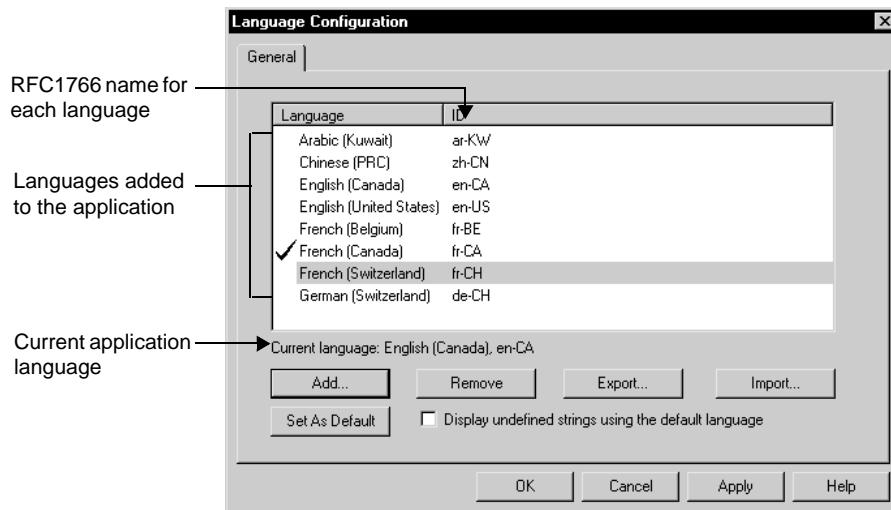
Use the Language Configuration dialog box in FactoryTalk View Studio to add up to 40 languages to the application.

Add languages before you:

- create language switch buttons.
- create the runtime application.

### To add languages to an application

1. On the Tools menu, click Languages.



2. Add languages. For details, see Help.

## Removing languages

You can also use the Language Configuration dialog box to remove languages. When you remove a language, all the strings for the language are deleted the next time you save the application. Saving a component deletes strings for the removed language from the component.

## Exporting application text strings for translation

There are two options for exporting text strings:

- one language at a time, saving the text in a tab-delimited text file in Unicode format. The file can be translated and then imported back into the application.
- all languages at once, exporting into a single Microsoft Excel spreadsheet. The strings in the spreadsheet can be translated and then imported back into the application.

These text strings allow language switching and are exported for translation:

- display titles for On Top displays (defined in the Display Settings dialog box)
- text graphic objects
- captions that you define for graphic objects, including objects in global object displays
- alarm, local, and information messages
- alarm trigger labels
- embedded time, date, and numeric variables

Undefined strings are exported as undefined strings, regardless of whether you select the option in the Language Configuration dialog box to display undefined strings using the default language. That option is used only for displaying strings during application development and at run time. It does not substitute the default language's strings for undefined strings in the exported file or spreadsheet.

## Exporting text in Unicode format

When you export text in Unicode format, these text strings are also exported:

- text descriptions and error messages for the Change Password window. These strings are exported if you use a password button in your application.

You can translate these strings, but you cannot change their language dynamically at run time the way you can for graphic object strings.

For the Change Password window, the runtime application will use the strings that are in the application when the runtime application is created. For example, if the current application language is French, and you have imported French text strings for the Change Password window, then the French strings will be compiled in the runtime application. If the operator opens the window, the strings will be displayed in French. If the operator changes to another language, the strings will still be displayed in French.

## Unicode file name format

The format for the exported Unicode text file name is

*<ApplicationName><HMIServerName><RFC1766>.txt*, where:

- *<ApplicationName>* is the name of the application.
- *<HMIServerName>* is the name of the HMI server containing the text strings you exported. This is always the same name as the application name.
- *<RFC1766>* is the RFC1766 name associated with the language selected for the export operation.

For details about the schema of exported Unicode text files, see page 12-11.

For a list of Windows languages and the corresponding RFC1766 names, see page F-1.

## Exporting text to a Microsoft Excel spreadsheet

FactoryTalk View supports Microsoft Excel versions 2000, 2002, and 2003. The exported file is in Microsoft Excel 5.0/95 Workbook format, with the extension .xls.



The maximum number of characters viewable in an Excel cell depends on the version of Excel and the font size. If the maximum is exceeded, the cell will be blank.

## Optimizing duplicate strings

When you export text to an Excel spreadsheet, you have the option of optimizing duplicate strings. FactoryTalk View analyzes the text and indicates repeated instances of the same string using a reference number. This way the translator only needs to translate the text once. When you import the translated strings from the spreadsheet into the application, FactoryTalk View copies the translated string into all the graphic objects that used the string originally.

## Excel spreadsheet file name format

The format for the exported spreadsheet file name is

*<ApplicationName>\_<ExportVersion>.xls*, where:

- *<ApplicationName>* is the name of the application.
- *<ExportVersion>* is an incremental number assigned to each successful export operation.

For details about the schema of exported spreadsheet files, see page 12-9.

For a list of Windows languages and the corresponding RFC1766 names, see page F-1.

## Exported language string file locations

Exported language files are saved at this default location:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\Strings  
(Windows 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\Strings  
(Windows XP or Windows Server 2003 R2)

You can specify another location if desired.

### To export application text

1. On the Tools menu, click Languages.
2. In the Language Configuration dialog box, select the application language for which to export text strings.  
If you are going to export strings for all languages defined for the application, it does not matter which language is selected in the Language Configuration dialog box.
3. Click Export.
4. Follow the instructions in the String Import Export Wizard.

For details about options in the String Import Export wizard, click Help.

If you export to a Unicode text file multiple times to the same location, previously exported versions are overwritten.

### Problems exporting

You do not have to check every text file created during an export to verify that text strings were exported correctly. If errors occur, or if you cancel the export while it is in progress, a message appears in the Diagnostics List, and in the FactoryTalk® Diagnostics log file.

In addition, errors are displayed when they occur in a log file called ExportErrors.txt, which is saved at this location:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\Strings  
(Windows 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\Strings  
(Windows XP or Windows Server 2003 R2)

Existing error log files are overwritten for each subsequent export operation that generates errors.

## Translating application text in Excel spreadsheet files

This section contains information about the format and schema of the exported Microsoft Excel spreadsheet file.

The spreadsheet contains a header row followed by a row for each text string in the application. There is a column for each language defined in the application. The default language is first, followed by the remaining languages in alphabetical order by RFC1766 name. All cells use the text format.



Do not delete the header row in the spreadsheet. FactoryTalk View requires this information when you import the translated text into your application. Similarly, do not change the cell formats.

If you select the option to optimize duplicate strings, there is a reference column to the left of each language column. Do not change the information in the reference columns.

## Translating application text in Unicode files

This section contains information about the format and schema of the exported Unicode text file. It also contains information about using Microsoft Excel or Windows Notepad to edit the file.



Because FactoryTalk View requires that parts of the text file remain the way they were exported, give the information in this section to the translator, to ensure that the file can be imported after it has been modified.

### File name and format

You will likely want to rename the file before translating it, to avoid confusing it with the original file. You can use any file name ending with the file extension .txt.

To import text into a FactoryTalk View application, you must save the file as tab-delimited text, in Unicode text format.

### Opening the text file in Microsoft Excel

When you open the text file in Microsoft Excel, the Text Import Wizard appears.

#### To specify the file format (Step 1 of the wizard)

1. Click Delimited.
2. In the Start import at row box, type or select 1.
3. In the File origin list, click Windows (ANSI).
4. Click Next.

### To specify the field delimiter (Step 2 of the wizard)

1. Select the Tab check box. If any other check boxes are selected, clear them.
2. Make sure the “Treat consecutive delimiters as one” check box is cleared.
3. Click Next.

### To specify the column data format (Step 3 of the wizard)

1. If it is not selected already, under Column data format, click General.
2. Click Finish.

## Saving the text file in Microsoft Excel

### To save the file

1. On the File menu, click Save.
- Excel warns that the file may contain features that are not compatible with Unicode text.
2. When prompted to keep the workbook in Unicode format that leaves out incompatible features, click Yes.
3. Close the file.
4. When prompted to save changes, click Yes.
5. When prompted again to keep the workbook in Unicode format that leaves out incompatible features, click Yes.

### Differences in file format for files saved in Excel

If you use Notepad to open a Unicode text file that was saved in Excel, you will notice some differences from a file edited and saved in Notepad.



You do not have to change the format of the file before you import it into FactoryTalk View.

The differences are:

- Double quotes surrounding the string definitions are removed for most strings.
- String definitions containing embedded double quotes or other characters that Excel treats as special characters, such as commas, are enclosed within double quotes.
- Any embedded double quotes are converted to a pair of double quotes.

## Saving the Unicode text file in Notepad

When saving the file, save it using the Unicode encoding option in the Save As dialog box.

### File schema

#### Comments

The text file uses the # symbol as a comment delimiter if it is the first character on a line.

#### Header

The first seven lines of the text file contain header information that must not be translated or modified.

#### Body

The body of the text file starts on line eight, and includes the following columns:

	Component type	Component name	String reference	“String definition”
	Graphic Display	Pump station	1	“Stop motor”

The file is sorted alphabetically by component name, and then numerically by string reference number.

Each string reference number refers to a different object in the component. In the example shown above, string reference 1 might refer to a momentary push button in the graphic display called Pump station.

The string definition is enclosed in quotes in Notepad, but not in the spreadsheet column in Excel.

In the translated text file, the only text that can be modified is the text inside the quotation marks in the string definition column. For example, translated into German, the file would contain these changes:

	Component type	Component name	String reference	“String definition”
	Graphic Display	Pump station	1	“Motor abschalten”

Do not change the entries in the component type or component name columns, unless the component was renamed in the application after the text was exported.



Do not modify the string reference number. The string reference number is a unique number that is assigned to an object by FactoryTalk View. Modifying the string reference number prevents FactoryTalk View from identifying the object correctly when you import the text.

## Working with pairs of double quotes

If a text string contains double quotes, the whole string definition must also be enclosed in double quotes. For example:

**Call "Duty Manager"**

must be entered in the string file as:

**"Call "Duty Manager""**

## Importing text containing multiple sets of double quotes

If the string definition contains an odd number of double quotes, the number of double quotes is rounded down to an even number, and then each pair is imported as one double quote. For example, the string:

**"Call "Duty Manager"**

appears in the application as:

**Call Duty Manager**

## Working with backslashes and new line characters

To force text to begin on a new line, precede the text with the characters **\n**. For example:

**Motor\nStopped**

appears in the application as:

**Motor**

**Stopped**

To make the characters **\n** appear as part of the text, type **\\\n**. To make a backslash appear in the application, type two backslashes (**\\\**).

## Importing text containing multiple backslashes

If the imported text file contains an odd number of backslashes next to each other, one of the backslashes will be ignored.

For example, the string:

**Seven\\|Eight**

is imported into the application as:

**Seven\\|Eight**

## Importing text

You can import text strings saved in Unicode text format or in Microsoft Excel spreadsheet format.

If your text strings are in Microsoft Excel spreadsheet format, only text strings that exist in the spreadsheet are imported. This means that existing strings in the application are not deleted during the import if they are not in the spreadsheet.

Back up the text in your application before importing. You can do this by exporting the text.



Save the backup file in a different location than the translated file you are about to import.

### To import text into your application from a text file

1. On the Tools menu, click Languages.
2. In the Language Configuration dialog box, select the application language for which to import text strings.

If you are going to import strings for multiple languages defined in an Excel spreadsheet, it does not matter which language is selected in the Language Configuration dialog box.

3. Click Import.
4. Follow the instructions in the String Import Export Wizard.

For details about options in the String Import Export wizard, click Help.

If you cancel the import before it is complete, any text strings that were changed are not restored to their original values. To restore the text that was originally in the application, import the text from the backup file you created.

## Problems importing

You do not have to check every graphic display in your application to verify that text was imported correctly. If errors occurred while importing text, they are displayed automatically from a log file called ImportErrors.txt in the following folder:

C:\Documents and Settings\All Users\Documents\RSView Enterprise\Strings  
(Windows 2000)

or

C:\Documents and Settings\All Users\Shared Documents\RSView Enterprise\Strings  
(Windows XP or Windows Server 2003 R2)

Each time errors occur while importing text into an application, the ImportErrors.txt file is overwritten.

If errors occurred while importing text, or if the import was canceled, a message appears in the Diagnostics List, and in the FactoryTalk Diagnostics log file.

## **Cancelling importing**

If some, but not all, of the text in an application seems to have been modified, the import might have been canceled. If you cancel the import before it is complete, any text strings that were changed are not restored to their original values. To restore the text originally in the application, import the text from the backup text file you created before importing.

## **Setting up multiple language support for graphic libraries**

When you create an application, the graphic libraries installed with FactoryTalk View are “language neutral” by default.

This means that graphic objects in the libraries always display their text strings as shipped, regardless of the current application language. Since the current application language is ignored, text strings never appear as undefined.

You can change the default for any graphic library, so that it supports multiple languages instead of a single language. When you do this, when you open the library you can view the text strings that are defined for the current application language. If the library’s text strings have not been defined for the current language, the undefined strings appear as single question marks.

If you turn off a graphic library’s multiple language support, and then save the library, only strings for the current application language are saved. The current application language therefore becomes the “language neutral” language. Any strings for other languages are deleted.

### **To turn on support for multiple languages in a graphic library**

1. Right-click an empty area in the graphic library display, and then click Display Settings.
2. In the General tab, select Support Multiple Languages.

After you turn on support for multiple languages, when you save the graphic library, all strings that support language switching are saved in the current application language. For information about saving libraries in multiple languages, see page 19-17.

## Using graphic libraries that support multiple languages

To use the graphics libraries in an application, you can add a graphic library into the Displays folder or copy objects from a graphic library into a graphic display. If the graphic library supports multiple languages:

- when you add the graphic library into the Displays folder, all strings, including strings for languages that are not supported by the application, are included with the display.
- if you copy an object from the graphic library into a graphic display, only strings for languages supported by the application are copied.

For more information about using graphic libraries, see Chapter 19.





# 13 Setting up display navigation

This chapter describes:

- what display navigation is.
- developing a hierarchy of displays.
- testing display navigation.
- using graphic objects to navigate.
- controlling display changes remotely.

This chapter describes methods for navigating between graphic displays. For information about navigating between objects in a graphic display, see page 21-7.

## About display navigation

The term display navigation refers to the way the operator moves between the graphic displays that make up an application.

Use these methods to set up display navigation for your application:

- Develop a hierarchy of graphic displays, to chart how users will navigate the application.
- Determine which users will have access to which parts of the application.
- Create graphic objects that the operator can use to navigate the application.
- Use the Remote Display Number connection to automatically control display changes. The use of this connection is optional. Assign it in the Global Connections editor.
- In the Startup editor, specify the graphic display to open when the application starts.
- Set up security so that only authorized users have access to the application or parts of the application.

For information about setting up security, see Chapter 11.

## Developing a hierarchy of displays

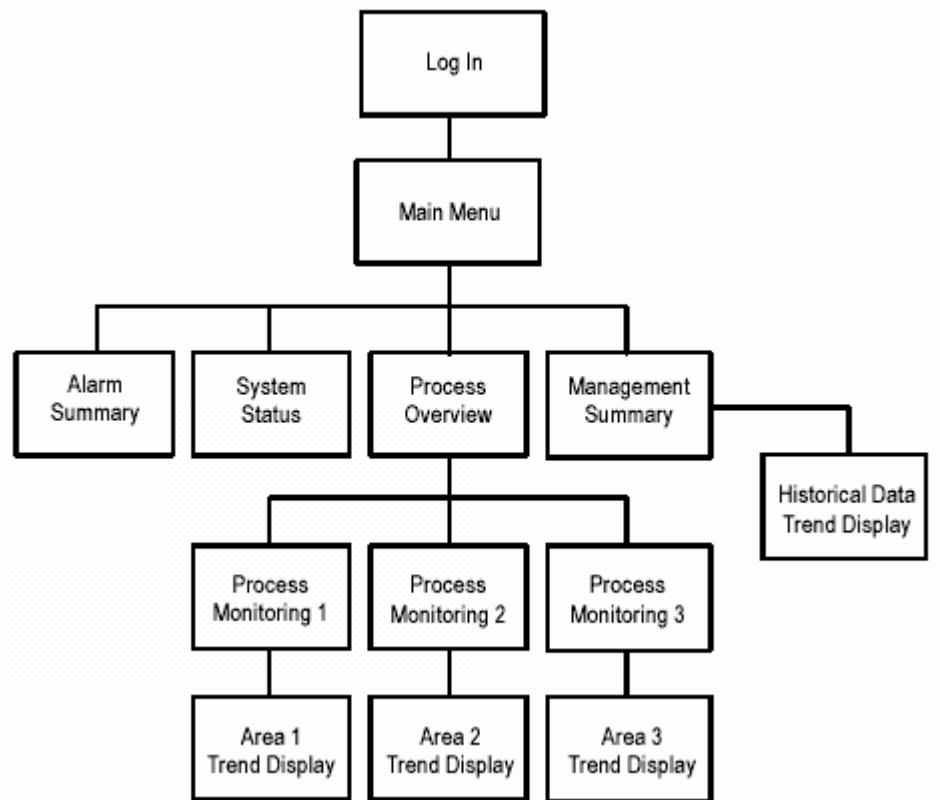
A display hierarchy is a series of graphic displays that provide progressively more detail as users move through them. Design your display hierarchy to meet the needs of the various users, including managers, supervisors, and operators.

If you plan to use security, determine which groups of users need access to which displays, and decide where in the hierarchy to locate login, logout, password, and shutdown buttons. For information about setting up security, see Chapter 11.

A display hierarchy could include:

- an initial graphic display for logging in.
- a graphic display that serves as a menu.
- an overview of the plant.
- a comprehensive display of each process being monitored.
- process-specific displays that provide more detail.
- management summary displays.
- trend displays of historical and current data.

The following illustration shows a sample display hierarchy.



## Testing display navigation

Once you set up display navigation for your application, test the application to make sure that navigation flows smoothly and that you have avoided problems like these:

- A graphic display contains no buttons for moving forward or back.
- When a graphic display closes, no other display is open and there is no way to continue using the application.

For information about testing your application, see page 14-2.

## Using graphic objects to navigate

Use these graphic objects to navigate through the displays in the application:

Use this object	To do this
Goto display button	Open the specified graphic display.
Goto configure mode button	Stop the application and open the FactoryTalk® View ME Station dialog box.
Return to display button	Close the current display and open the previous display.
Close display button	Close the current display. Can send a value to a tag when the display closes.
Display list selector	Provide a list of graphic displays so the operator can select which display to open.
Shutdown button	Stop the application and exit FactoryTalk® View ME Station.

The behavior of these graphic objects depends on which types of graphic displays are used. See the following sections for details.

For detailed information about setting up graphic objects, see Help.

## Switching languages

If the application uses multiple languages and operators will be switching languages at run time, place language switch buttons in a display that the operators have access to. For example, put the buttons in the display that opens when the operators log in. Create a language switch button for each language that the operators will be using.

For more information about setting up multiple languages, see Chapter 12.

## Display type

The display type you use gives you additional control over how the operator navigates between displays. For example, use the On Top type to keep a graphic display on top at all times, even when another display has focus. Or use the Replace type if you want a display to replace all other open displays when it opens.

For more information about display types, see page 19-12.

## Goto display buttons

When you set up a goto display button, specify the graphic display to open when the operator presses the button at run time.

The specified display doesn't open if the operator does not have security access for the display.

You can also assign a parameter file that assigns tags to tag placeholders in the display when the display opens. For more information about parameter files, see page 25-3.

For information about setting up goto display buttons, see Help.

## How display types affect the button's behavior

The goto display button's behavior at run time also depends on which types of graphic displays are already open and which type of display it is opening:

- If the graphic display assigned to the button is a Replace display, it closes any open On Top or Replace displays. It does not close any On Top displays that use the Cannot Be Replaced option.
- If the display assigned to the button is already open, but does not have focus, pressing the button gives the display focus.

The operator cannot use the goto display button to open Replace displays if display change is currently controlled remotely (using global connections). But the operator can still open On Top displays.

## Goto configure mode buttons

When the operator presses a goto configure mode button at run time, the current application stops running and the FactoryTalk View ME Station dialog box opens.

While in configure mode, the operator can use FactoryTalk View ME Station to change applications, application settings, and terminal settings. The operator can also delete log files. For more information about the FactoryTalk View ME Station dialog box, see Chapter 15.

## Return to display buttons

When the operator presses a return to display button at run time, the graphic display that the button is on closes and the display that was previously open reopens.

The current display does not close if:

- the display change is currently controlled remotely (using global connections).
- there were no previously opened Replace displays.
- the operator does not have security access for the previous display. This can only occur if a new user logs in using a login button in the current display.

The return to display button only goes back to the most recent display. It doesn't go back through a series of displays.

For information about setting up return to display buttons, see Help.

## How display types affect the button's behavior

Both the current display and the previous display must be Replace displays. When the operator presses the return to display button:

- If the graphic display that is closing is a Replace display, the display closes and the previously opened Replace display opens. Any On Top displays that were previously open with the Replace display are not reopened.
- If the graphic display that is closing is an On Top display, the display closes but no display is reopened.

We therefore recommend that you use return to display buttons in Replace displays only.

---

## Example: Navigating through displays

This example uses the display hierarchy illustrated on page 13-2, and shows what happens as the operator navigates through the hierarchy. The graphic displays are all Replace displays.

1. In the Main Menu display, the operator uses a display list selector to open the Process Overview display.
2. In the Process Overview display, the operator presses a goto display button to open the Process Monitoring 1 display.
3. After viewing the state of the process, the operator presses a return to display button to close the current display and reopen the Process Overview display.

4. In the Process Overview display, the operator presses a return to display button. Which display opens? The Process Monitoring 1 display (because this was the previously opened display).

To return to the Main Menu display from the Process Overview display, the operator would have to press a goto display button that is set up to open the Main Menu display.

---

### **Close display buttons**

When the operator presses a close display button at run time, the graphic display that the button is on closes. You can set up the button to write out a value when the display closes.

If the graphic display that is closing is a Replace display, and the display change is controlled remotely, the display does not close. If the display does not close, the close value, if any, is not written out.

If the display change is controlled by the operator and the graphic display that is closing is a Replace display, if there are no On Top displays open, an empty window is displayed. The operator will not be able to use the application again (unless a remote display change occurs or an alarm, activity, or information message display opens).

We therefore recommend that you use close display buttons in On Top displays only.

For information about setting up close display buttons, see Help.

### **Display list selectors**

Use the display list selector to show a list of graphic displays that the operator can choose from. The operator can scroll through the list and select the graphic display to open.

The specified display doesn't open if the operator does not have security access for the display.

You can also assign a parameter file that assigns tags to tag placeholders in the display when the display opens. For more information about parameter files, see page 25-3.

For information about setting up display list selectors, see Help.

## How display types affect the selector's behavior

The display list selector's behavior at run time also depends on which types of graphic displays are already open and which type of display it is opening:

- If the selected graphic display is a Replace display, it closes any open On Top and Replace displays. It does not close On Top displays that use the Cannot Be Replaced option.
- If the selected display is an On Top display, it opens on top of the current display. The current display does not close.

The operator cannot use the display list selector to open Replace displays if display change is currently controlled remotely. But the operator can still open On Top displays (with or without the Cannot Be Replaced option).

## Selecting the display to open

The operator can scroll through the list and select displays using the key button graphic objects, or, if the list has the input focus, by using the arrow keys and Enter key on a keypad or external keyboard.

You can link key buttons to a specific display list selector, or set up the buttons to work with whichever object is selected in the graphic display.

For information about input focus, see page 21-7. For information about linking buttons to the display list selector, see page 21-9.

## Shutdown buttons

When the operator presses the shutdown button at run time, the application stops and FactoryTalk View ME Station closes.

To prevent an unauthorized user from stopping the application, assign visibility animation to the shutdown button. For details, see page 11-14. Or, place the button in a display that only authorized users have access to.

For information about setting up shutdown buttons, see Help.

## Controlling display changes remotely

To control display changes remotely, you can set up the data source to open graphic displays using global connections.

Global connections are connections that apply to your entire runtime application. Global connections allow the data source to control or interact with your application at run time.

For example, the Remote Display Number connection is a global connection that you can use to control display changes from the data source. You can also use global connections

- 
- 
- FACTORYTALK VIEW MACHINE EDITION USER'S GUIDE
- 
- 

to print graphic displays from the data source, to run macros from the data source, and to control the date and time displayed on the runtime terminal. For more information about global connections, see Chapter 8.

# 14 Creating runtime applications

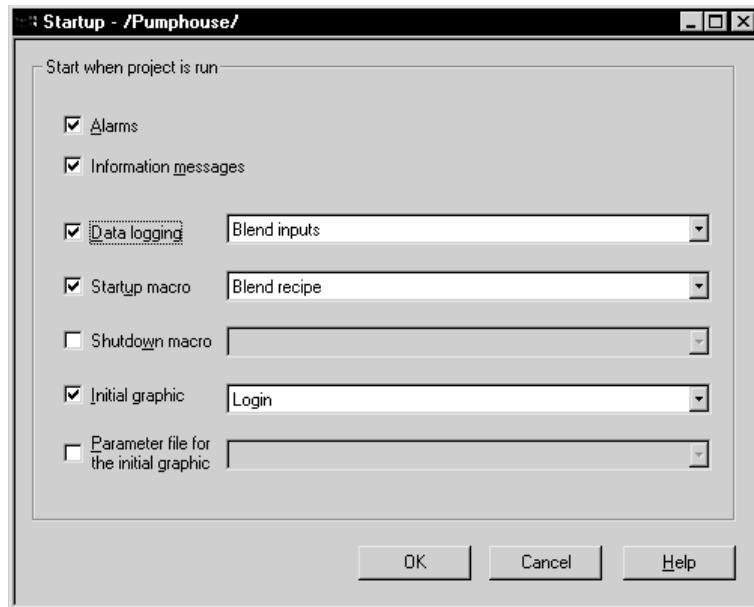
This chapter describes:

- specifying startup settings.
- testing your application.
- creating runtime application files.
- converting runtime application files to development applications.

## Specifying startup settings

Use the Startup editor to specify which application processes and components to start when the application starts at run time.

You can specify startup settings once you've set up all the parts of the application, or you can specify processes and select components in the Startup editor as you create them.



For detailed information about the options in the Startup editor, see Help.

## Testing your application

You can test your application in FactoryTalk® View Studio at any time during the development process, to make sure that everything works the way you intend.

If the development computer is connected to the data source, you can test all functions of the application, including security settings, language switching, communications, and alarm monitoring.

A FactoryTalk View ME Station emulator opens on the development computer and runs the application. This runtime version of the application is a temporary version for testing use only. You cannot run it on another computer.

There is a two-hour time limit for test running the application in FactoryTalk View Studio.

The procedure in this section shows you how to test your entire application. For information about testing a single graphic display, see page 19-10.

### To test your application in FactoryTalk View Studio



Test Application tool

1. On the Application menu, click Test Application, or click the Test Application tool.
2. If your application uses multiple languages, specify the languages to include and the initial runtime language, and then press Finish. For details, see Help.
3. Test your application.
4. To stop your application, press a shutdown button, or type the character ‘x’.

Make sure you provide the operator with a method for shutting down the application at run time. For more information about methods for shutting down applications, see page 15-5.

Once you've tested your application to make sure everything works the way you intend, create the runtime application file and transfer the file to the runtime computer.

## Creating runtime application files

Before you can run your application, you must create a runtime version. When you create the runtime version, FactoryTalk View Studio compiles all of the necessary application information into a single file with the extension .mer.

### Creating .mer files for previous versions

You can specify the version of FactoryTalk View ME Station for which to create the .mer file. For example, if the application will run on a terminal that uses FactoryTalk View ME Station version 3.20, you can specify that version for the .mer file.

If the application contains features that are not supported by the version you select, FactoryTalk View displays a validation report that lists the unsupported features. The

runtime application file is not created. You must remove or turn off the unsupported features before you can create the runtime application file.

For information about the features supported in different versions of FactoryTalk View, and how to remove or replace them, see Appendix G.

## Converting .mer files to development applications

For FactoryTalk View version 5.00 .mer files, you can convert the runtime .mer file back to a development application (.med file). This option is useful if:

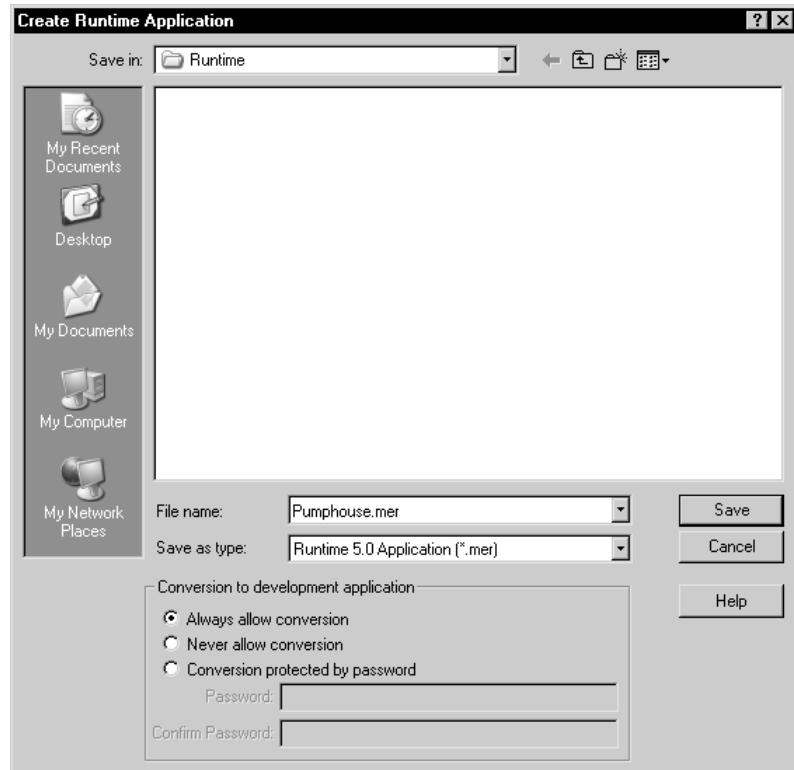
- you lose or don't have access to the original .med file and you need to make a change to the application.
- you make changes to the .mer file at run time, such as adding passwords, updating RSLinx® Enterprise™ device shortcuts, or saving recipes, and you need to make a change to the application but don't want to lose this runtime information.
- you need to extract information that was generated at run time, such as updated RecipePlus files.

When you create the .mer file, you can specify whether or not to allow conversion of the runtime .mer file, and whether to require a password to convert the runtime application.

For more information about converting .mer files to development applications, see page 14-6.

### To create a runtime application

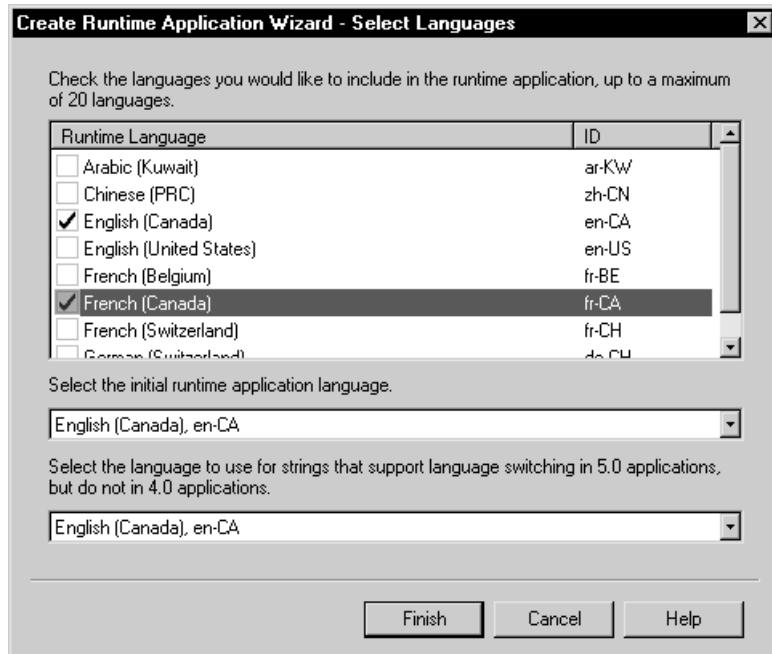
1. In FactoryTalk View Studio, with the application open, on the Application menu click Create Runtime Application.



2. Specify the folder and file name for the runtime application.
3. In the Save as type box, specify the version of FactoryTalk View ME Station for which to create the .mer file.
4. For version 5.00 .mer files, if you don't want to allow the runtime application to be converted to a development application, click Never allow conversion.
5. For version 5.00 .mer files, if you require that the user enter a password to convert the application, click Conversion protected by password. Type the password in the two boxes.

The password can be up to 100 characters long. The password is case sensitive.

6. Click Save.



7. If your application uses multiple languages, specify up to 20 languages to include in the runtime application, specify the initial runtime language, and specify the language to use for alarm trigger labels and alarm, information, and local messages in FactoryTalk View 4.00 runtime applications. For details, see Help.

This wizard is not displayed if your application uses only one language.

To display undefined strings in the default language at run time, include this language in your selection. The default language is also required for filtering alarms. For more information about the default language, see page 12-2.

8. Click Finish.

For information about transferring the runtime application:

- to a personal computer, see Chapter 15.
- to a PanelView™ Plus or PanelView™ Plus CE terminal, see Chapter 16.

## Converting runtime application files to development applications

For FactoryTalk View version 5.00 .mer files, you can convert the runtime application file back to a development application. This allows you to make changes to the application without losing runtime information such as passwords and RecipePlus data.

To convert the runtime application file, use the Restore runtime application operation in the Application Manager. Data logging and alarm data that occurred at run time are not restored to the development application.

If your application uses FactoryTalk Security, you must have write, backup, and restore permissions to convert the runtime application to a development application. For more information about FactoryTalk Security, see Chapter 11.

If desired, you can also restore the runtime application's FactoryTalk Local Directory to the development computer. The development computer's FactoryTalk Local Directory will be backed up, and the runtime version will replace it. If you choose this option, make sure you have access to the runtime FactoryTalk Local Directory. For more information about the FactoryTalk Local Directory, see Chapter 11.

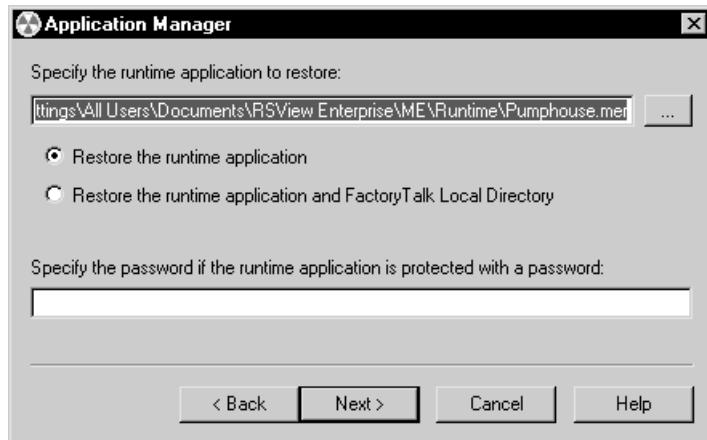
RSLinx Enterprise device shortcuts that were edited at run time are restored when you convert the runtime application to a development application. However, the Runtime tab in the RSLinx Enterprise Communication Setup editor might not display the device configuration for the runtime application. Make sure the Runtime tab shows the correct device configuration before making any changes to the shortcuts.



RSLinx Enterprise device shortcuts that were valid on the runtime terminal might not exist or might be invalid on the development computer. Review all device and tag file shortcut assignments in the RSLinx Enterprise Communication Setup editor before you edit, test, or transfer the application.

### To convert a runtime application to a development application

1. In FactoryTalk View Studio, on the Tools menu, click Application Manager.
2. When prompted to select the application type, click Machine Edition, and then click Next.
3. Click Restore runtime application, and then click Next.



4. Specify the folder and file name of the runtime application to convert.
5. To restore the runtime application's FactoryTalk Local Directory to the development computer, click Restore the runtime application and FactoryTalk Local Directory.
6. If you created a password for the file when you created the runtime application, type the password.
7. Click Next.
8. Type a name for the converted application, and then click Finish.





# 15 Running applications on a personal computer

This chapter describes:

- moving applications to the runtime computer.
- starting FactoryTalk® View ME Station.
- loading and running applications.
- shutting down applications.
- editing device short cuts.
- looking up contact information for technical support.
- setting up FactoryTalk® Diagnostics at run time.
- setting up serial ports for KEPServerEnterprise.
- setting up RSLinx® Enterprise™ communication drivers.
- specifying the printers to use at run time.
- specifying startup options for FactoryTalk View ME Station.
- deleting log files on the runtime computer.
- specifying the time, date, and number formats to use at run time.
- using the DeskLock tool.

## Summary of steps

Follow these steps to:

- install the necessary hardware and software on the runtime computer.
- transfer your Windows® 2000, Windows XP, or Windows Server 2003 R2 application to the runtime computer.
- set up options in FactoryTalk View ME Station.

For information about installing FactoryTalk View ME Station, see the *FactoryTalk View Machine Edition Installation Guide*.

## Installing hardware and software on the runtime computer

1. If you will be printing displays, alarms, or diagnostics messages, set up printer connections on the runtime computer.

For more information, see page 15-10.
2. If you are using RSLinx Enterprise, set up communications as described in Chapter 5.
3. If you are using RSLinx® Classic™ as the OPC® server on the runtime computer, install RSLinx Classic on the runtime computer.
4. If you are using RSLinx Classic on a remote computer, install RSLinx Classic on the remote computer.
5. If you are using an OPC server other than RSLinx Enterprise or RSLinx Classic, install the OPC server software on the runtime computer or on another computer on the network.

For installation information, see the documentation supplied by your OPC server vendor. For information about OPC, see Chapter 5.

6. If your application uses third-party ActiveX® objects, install and register the Windows 2000, Windows XP, or Windows Server 2003 R2 version of the objects on the runtime computer.

For information about ActiveX objects, see page 20-21.
7. Install on the runtime computer all languages used by the runtime application.
8. If the runtime computer uses different time, date, or number formats than the development computer, specify the time, date, and number formats to use at run time.

For more information, see page 15-16.
9. If desired, use the DeskLock tool to prevent users from switching to another software application or using the Windows desktop at run time.

For more information, see page 15-17.

## Transferring the application

- Move the application to the runtime personal computer.

For more information, see page 15-3.

## Setting up options in FactoryTalk View ME Station

1. On the runtime computer, start FactoryTalk View ME Station.

For more information, see page 15-4.

2. Load the application.  
For more information, see page 15-4.
3. Edit device shortcuts, if necessary.  
For more information, see page 15-6.
4. Set up FactoryTalk Diagnostics on the runtime computer (if you have not already done so).  
For more information, see page 10-6.
5. If you will be using KEPServerEnterprise™, specify serial port IDs.  
For more information, see page 15-8.
6. If you will be using RSLinx Enterprise, set up communication drivers (if you have not already done so).  
For more information, see page 15-9.
7. Specify the printers to use.  
For more information, see page 15-10.
8. Specify startup options for FactoryTalk View ME Station.  
For more information, see page 15-11.

Once you've completed these steps, you're ready to run the application. For information about running your application, see Chapter 17.

## **Moving applications to the runtime computer**

The runtime application file has the extension .mer. You can use any standard file transfer method to copy your runtime application from the development computer to the runtime computer.

You can:

- copy the application file from the development computer to a floppy disk, and then from the floppy disk to the runtime computer.
- if the application file is too large to fit on a floppy disk, use a larger storage device such as a Zip® disk.
- if the development and runtime computers are on the same network, use Windows Explorer or My Computer to move the file.

For information about creating the runtime application file, see Chapter 14.

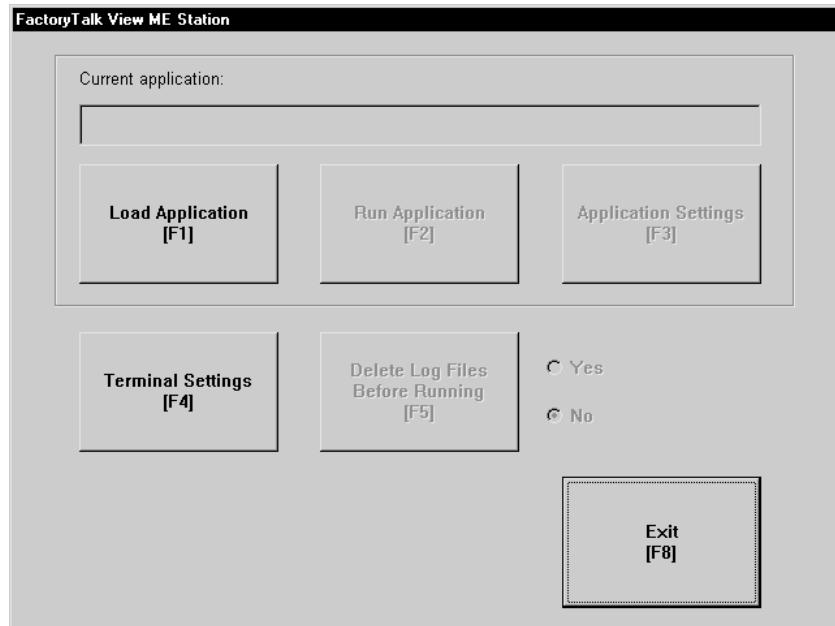
## Starting FactoryTalk View ME Station

If you are running an application on the development computer, we recommend that you exit FactoryTalk View Studio before starting FactoryTalk View ME Station.

### To start FactoryTalk View ME Station

1. On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk View, and then click FactoryTalk View ME Station.

The FactoryTalk View ME Station dialog box opens.



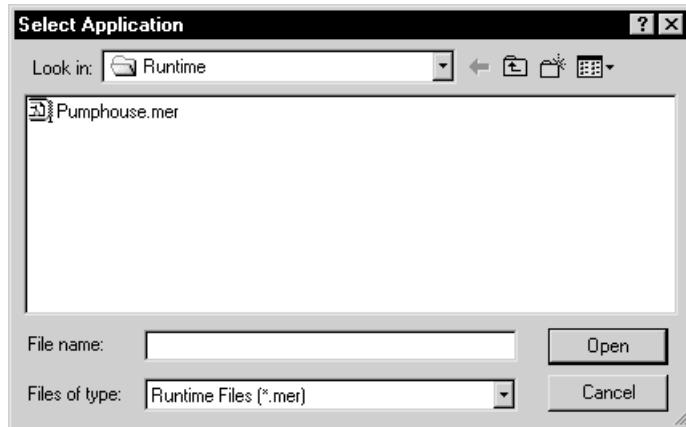
For information about specifying startup options for when FactoryTalk View ME Station starts, see page 15-11.

### Loading and running applications

You can run any runtime application that is on the runtime computer. Runtime applications have the extension .mer.

#### To load and run the application

1. In the FactoryTalk View ME Station dialog box, click Load Application, or press F1.



2. Navigate to the folder containing the application's .mer file, and then click the file name.
3. Click Open.
4. To replace the runtime computer's communication settings with the application's communication settings, click Yes. To keep the runtime computer's communication settings, click No.

You are notified that the application's FactoryTalk System Directory of users and security policies will be loaded on this computer. This is the set of users and policies that have been set up for the application and are contained in the .mer file.

The computer's existing FactoryTalk System Directory will be archived, and will be restored when you stop the application. To turn off this warning, see page 15-16.

5. To continue, click Yes. To stop loading the application, click No.

If you continue, the application name is displayed in the Current application box in the FactoryTalk View ME Station dialog box.

6. To run the application, click Run Application.

The DEFAULT user is logged in. If a macro is assigned to the DEFAULT user, the macro runs.

## Shutting down applications

### To shut down an application, use one of these methods

- Press a shutdown button in a graphic display.

- If the application is set up to use a title bar with a Control box, click the Close button at the right end of the title bar.
- If the application is set up to use a title bar with a Control box, on the Control menu at the left end of the title bar, click Close.



For information about using a title bar in graphic displays, see page 4-14. For information about preventing unauthorized users from shutting down applications, see page 11-14.

## **What happens when the application shuts down**

When the application shuts down:

- if you assigned a shutdown macro (in the Startup editor), the macro runs, assigning values to tags, and then the application stops.  
For information about the Startup editor, see Chapter 14.
- FactoryTalk View ME Station closes.
- The computer's FactoryTalk System Directory is restored.

## **Changing application settings**

### **Editing device shortcuts**

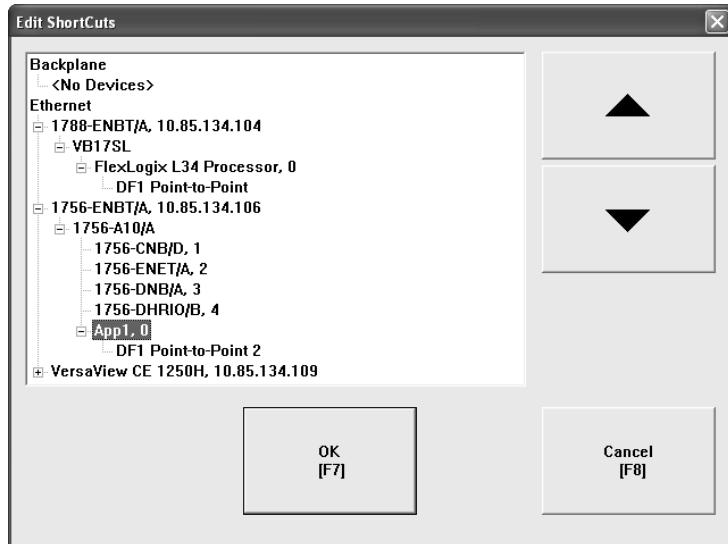
You can use FactoryTalk View ME Station to edit device shortcuts that have been set up in the application.

Before editing device shortcuts, load the application containing the device shortcut, as described on page 15-4.

### **To edit device shortcuts**

1. In the FactoryTalk View ME Station dialog box, click Application Settings.
2. Double-click Device Shortcuts.

3. Double-click the name of the shortcut to edit.



4. In the Edit ShortCuts dialog box, click the device you want the shortcut to point to, and then click OK.

## Looking up contact information for technical support

### To look up technical support contact information

1. In FactoryTalk View ME Station, click Terminal Settings.
2. Double-click About FactoryTalk View ME Station.
3. Click Technical Support.

The telephone number and URL for technical support are displayed.

## Setting up FactoryTalk Diagnostics on the runtime computer

You can set up FactoryTalk Diagnostics on the runtime computer using the FactoryTalk View ME Station dialog box.

### To set up FactoryTalk Diagnostics on the runtime computer

1. In FactoryTalk View ME Station, click Terminal Settings.
2. Double-click Diagnostics Setup.

The FactoryTalk Diagnostics Setup dialog box opens.

3. Set up FactoryTalk Diagnostics, as described on page 10-6.

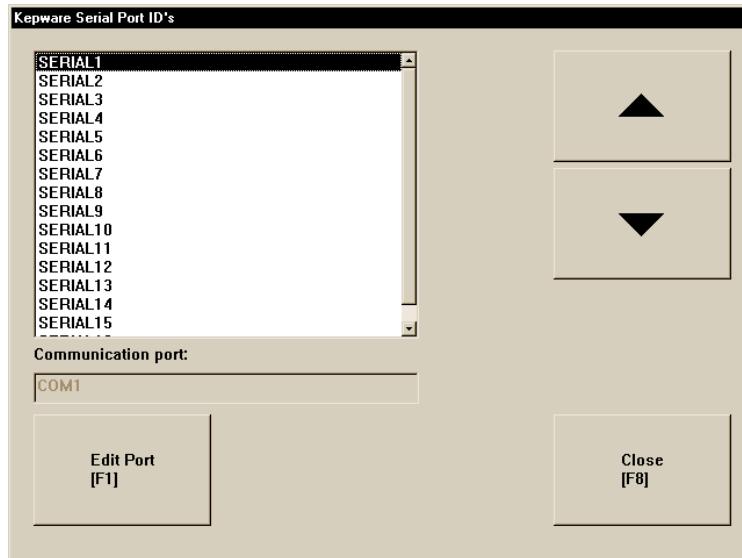
## Setting up serial ports for use with KEPServerEnterprise

If you plan to use KEPServerEnterprise and serial communications, you must specify which COM port to use.

For information about setting up communications in KEPServerEnterprise, see KEPServerEnterprise Help.

### To specify the COM port to use for serial communications

1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click Networks and Communications.
3. Double-click KEPServer Serial Port ID's.



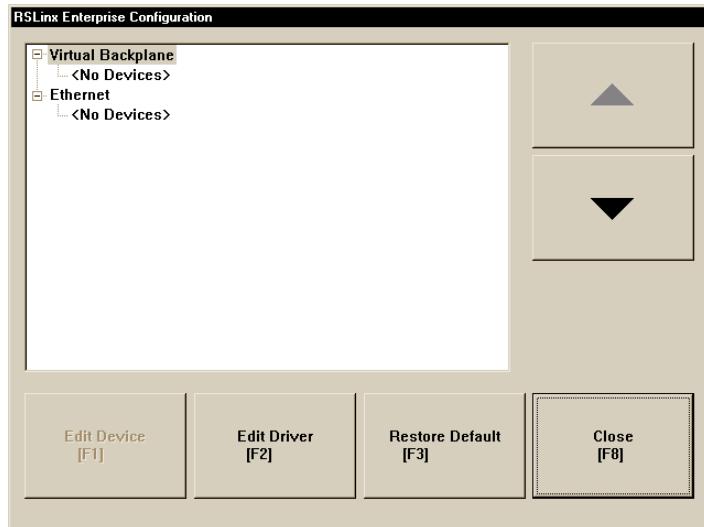
4. In the Kepware Serial Port ID's dialog box, click the serial port ID you specified when you set up the KEPServerEnterprise channel.
5. Click Edit Port.
6. In the Communication Ports dialog box, click the COM port to use for KEPServerEnterprise communications.

## Setting up RSLinx Enterprise communication drivers

Use RSLinx Enterprise to set up communication drivers for your runtime application. You can set up the drivers directly in RSLinx Enterprise, or open RSLinx Enterprise by using the FactoryTalk View ME Station dialog box.

### To set up the RSLinx Enterprise communication driver to use at run time

1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click Networks and Communications.
3. Double-click RSLinx Enterprise Communications.



4. To edit a driver, select it and then click Edit Driver.
5. To edit a device, select it and then click Edit Device.

For information about setting up RSLinx Enterprise drivers and devices, see the RSLinx documentation.

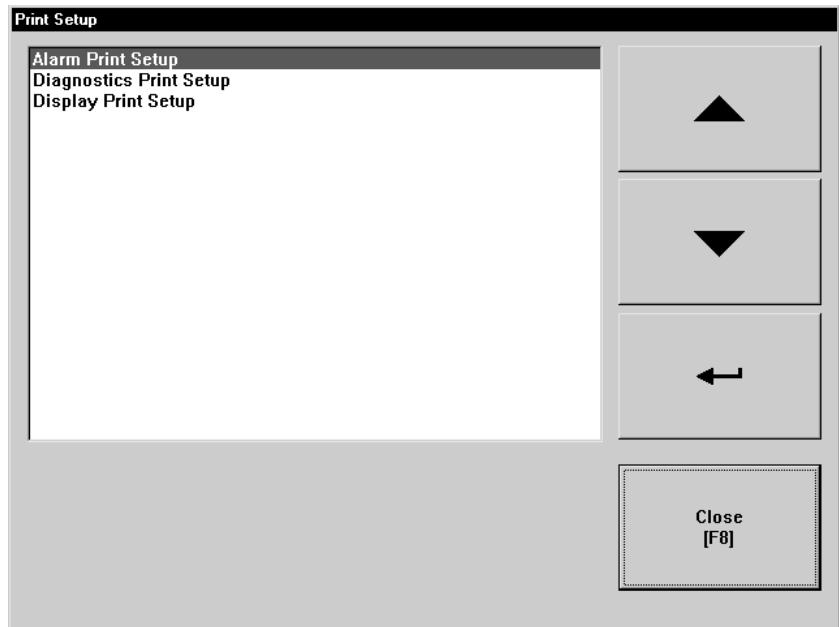
Once the driver is set up, FactoryTalk View ME Station automatically starts the driver software when you run the application.

## Specifying the printers to use at run time

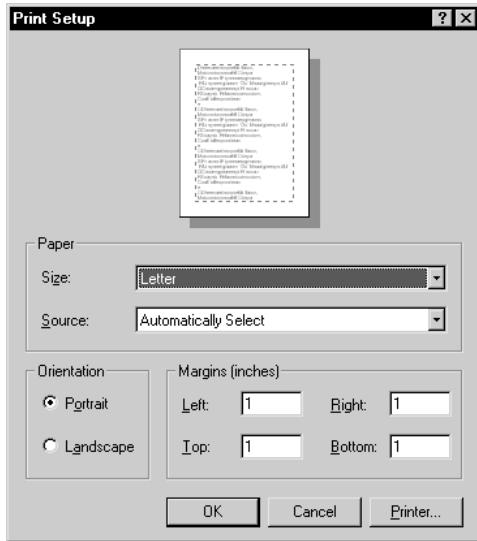
You can use local or network printers to print alarm messages, reports, diagnostics messages, and graphic displays at run time. If desired, you can use a different printer for each type of printing.

### To specify the printers to use at run time

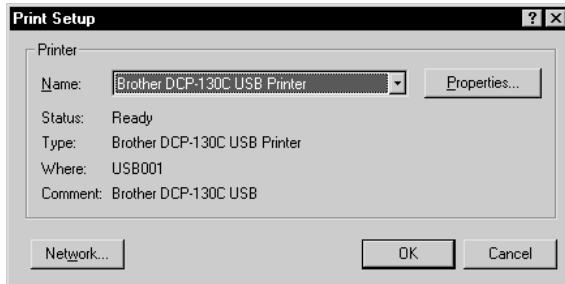
1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click Print Setup.



3. Double-click the type of printing to set up.



4. Click Printer.



5. Modify the print options as required.

For detailed information about printer options, refer to your Windows documentation.

## Specifying startup options for FactoryTalk View ME Station

This section describes how to start FactoryTalk View ME Station automatically when Windows starts, and describes startup options for FactoryTalk View ME Station.

When you start FactoryTalk View ME Station, you can:

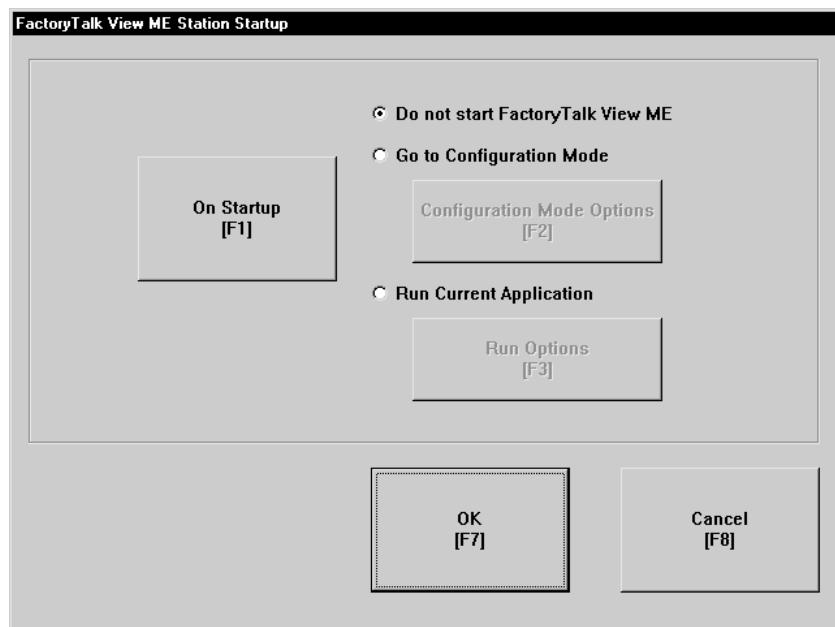
- run an application.

- load an application.
- automatically delete an application's log files before running the application.
- replace RSLinx Enterprise communications on the runtime computer with the application's settings.

By default, the option to start FactoryTalk View ME Station when Windows starts is turned off. The settings in this section apply only if you want FactoryTalk View ME Station to start automatically when Windows starts.

### **To start FactoryTalk View ME Station when Windows starts and run an application**

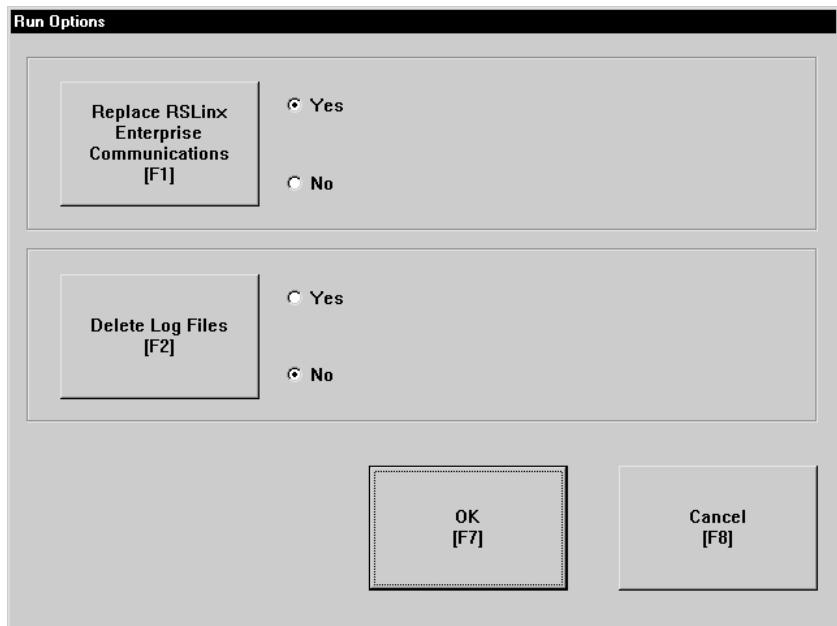
1. In the FactoryTalk View ME Station dialog box, load the application that you want to run.  
For details, see page 15-4.
2. Click Terminal Settings.
3. Double-click FactoryTalk View ME Station Startup.



4. Click Run Current Application.

This option is not available if you have not loaded an application yet.

5. Click Run Options.

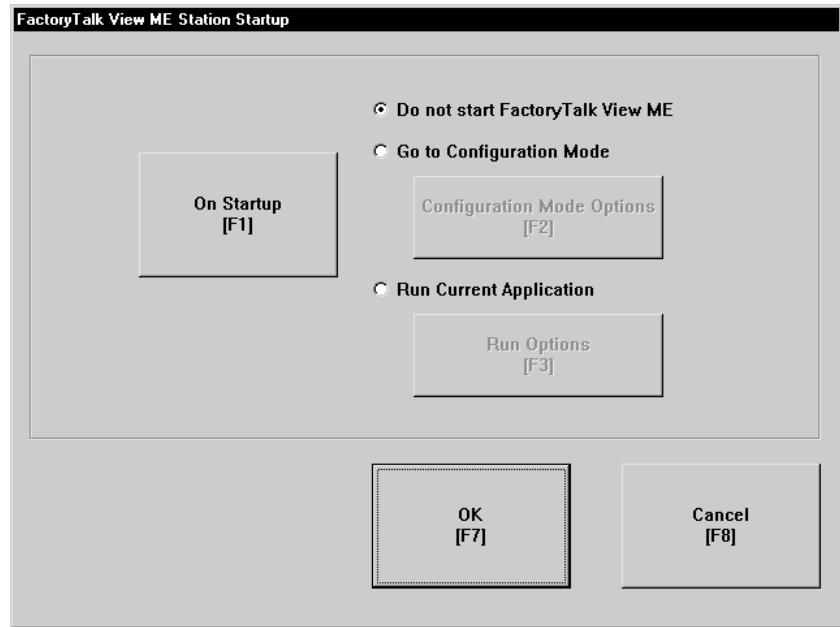


6. Specify whether to replace the RSLinx Enterprise communication settings on the runtime computer with the application's settings when the application starts.
7. Specify whether to delete the application's log files on startup.

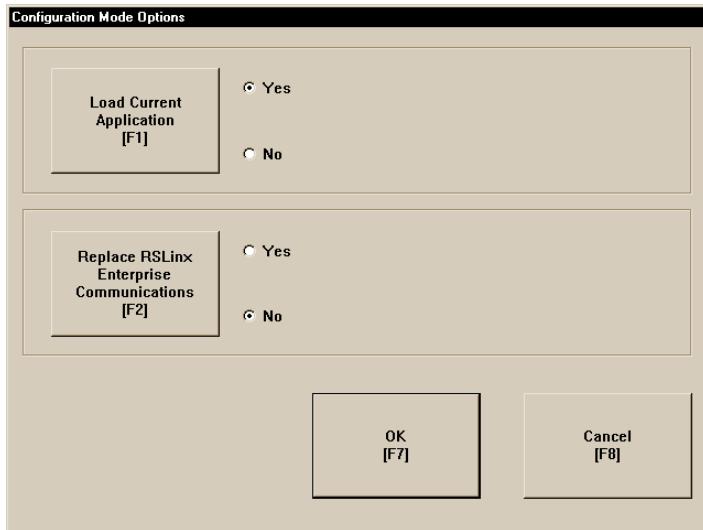
**To start FactoryTalk View ME Station when Windows starts without running an application**

1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click FactoryTalk View ME Station Startup.

- 
- 
- FACTORYTALK VIEW MACHINE EDITION USER'S GUIDE
- 
- 



3. Click Go to Configuration Mode.
4. Click Configuration Mode Options.



5. Specify whether to load the current application when FactoryTalk View ME Station starts.

This option is not available if you have not loaded an application.

6. Specify whether to replace the RSLinx Enterprise communication settings on the runtime computer with the application's settings when the application starts.

This option is not available if you have not loaded an application.

## **Deleting log files on the runtime computer**

When you run your application, FactoryTalk View ME Station stores log files for alarms and data logging (if you use these features). When you start FactoryTalk View ME Station, you can delete the alarm and data log files for the loaded application, or for all the applications on the runtime computer.

## **Running a newer version of the application**

If you run a newer version of an application, the alarm log file for the older version is deleted automatically. The data log file for the older version is retained, to allow the display of historical data in trends.

For more information about the alarm log file, see page 9-10. For more information about data log files, see page 26-5.

## **Deleting log files manually**

### **To delete log files for the loaded application**

1. In the FactoryTalk View ME Station dialog box, click Yes beside the Delete Log Files Before Running button.

All alarm and data log files for the loaded application are deleted.

### **To delete log files for all applications on the runtime computer**

1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click Delete Log Files.
3. Click Yes.

Log files for data log models that use the default path are deleted. All alarm log files are deleted.

## **Turning off the FactoryTalk Directory Server warning**

When you load an application, you are notified that the application's FactoryTalk System Directory of users and security policies will be loaded on the computer. The computer's existing FactoryTalk System Directory is archived while you run the application. It is restored when you stop the application.

You can turn off this notification warning.

### **To turn off the overwrite warning**

1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click System Directory Overwrite Warning.
3. Click No.

## **Specifying time, date, and number formats**

Time, date, and number formats are used by these graphic objects:

- numeric display
- gauge
- time and date display
- trend
- alarm banner
- alarm list

The time and date formats are also used when printing the alarm history report (using the print alarm history button). The numeric and time and date embedded variables also use number, time, and date formats.

All objects use the time, date, and number formats of the current application language. For example, if the application language uses a comma for the decimal symbol, numeric variables use a comma for the decimal symbol.

For information about using multiple languages, see Chapter 12.

## Using the DeskLock tool

Use the DeskLock tool to prevent users from switching to another software application or having access to the desktop at run time.

This tool can have far-reaching effects on your operating system. The DeskLock tool can replace the standard Windows 2000, Windows XP, or Windows Server 2003 R2 non-configured desktop with a customized desktop. The customized desktop is intended to prevent operators from having access to other applications and operating system functions such as restarting Windows or shutting down tasks. You can also use the tool to set up an automatic Windows logon and to specify options such as which software applications are allowed or not allowed to run on the computer.



Before using the tool, read the DeskLock Help completely.

### To open the DeskLock tool

1. On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk View, Tools, and then click DeskLock.

For details about using the tool, see the tool's Help.





# 16 Transferring applications to a PanelView Plus or PanelView Plus CE terminal

This chapter describes:

- starting FactoryTalk® View ME Station.
- setting up communication drivers to use to transfer applications to a PanelView™ Plus or PanelView Plus CE terminal.
- transferring applications and TrueType™ fonts to PanelView Plus or PanelView Plus CE terminals.
- uploading applications from a terminal to the development computer.
- comparing applications on the development computer with applications on the terminal.

For information about using your application at run time, see Chapter 17.

For information about using FactoryTalk View ME Station on the terminal, including setting up communications, printers, and input devices, see the *PanelView Plus Terminals User Manual*. This manual is available on the FactoryTalk View Machine Edition CD.

## Summary of steps

Follow these steps to:

- install hardware and software on a PanelView Plus or PanelView Plus CE terminal.
- transfer your application to the PanelView Plus or PanelView Plus CE terminal.

For information about installing FactoryTalk View ME Station, see the *FactoryTalk View Machine Edition Installation Guide*.

### Installing hardware and software on a PanelView Plus CE terminal

1. If you will be printing displays, alarms, reports, or diagnostics messages, connect a printer to the USB or Network port.

PanelView Plus CE terminals support printing using the Microsoft® Windows® CE PCL3 printer driver, which is already installed on the terminal. Printing is supported for most laser and ink-jet printers. If you have problems printing, check that your printer is compatible with the PCL3 printer driver.

2. If you are using an OPC® server other than RSLink® Enterprise™, for example, KEPServerEnterprise™, install the OPC server software on the terminal.

For installation information, see the documentation supplied by your OPC server vendor. For information about OPC data servers, see Chapter 5.

3. If your application uses third-party ActiveX® objects, install and register the Windows CE version of the objects on the terminal.

For information about ActiveX objects, see page 20-21.

### **Installing printers on a PanelView Plus terminal**

1. If you will be printing displays, alarms, reports, or diagnostics messages, connect a printer to the USB or Network port.

PanelView Plus terminals support printing using the Microsoft Windows CE PCL3 printer driver, which is already installed on the terminal. Printing is supported for most laser and ink-jet printers. If you have problems printing, check that your printer is compatible with the PCL3 printer driver.

The PanelView Plus terminal is a “closed box,” which means you must order any software your application will need when you order the system. All software comes pre-installed.

### **Transferring applications**

1. On the PanelView Plus or PanelView Plus CE terminal, start FactoryTalk View ME Station.

For more information, see page 16-3.

2. In FactoryTalk View ME Station, if you’re transferring via a serial connection, specify and set up the serial driver. If you are using Ethernet® for the transfer, you can skip this step.

For more information, see page 16-3.

3. On the development computer, set up the RSLinx Enterprise driver to use for the transfer.

For more information, see page 16-5.

4. Move the application to the PanelView Plus or PanelView Plus CE terminal using the Transfer Utility tool.

For more information, see page 16-5.

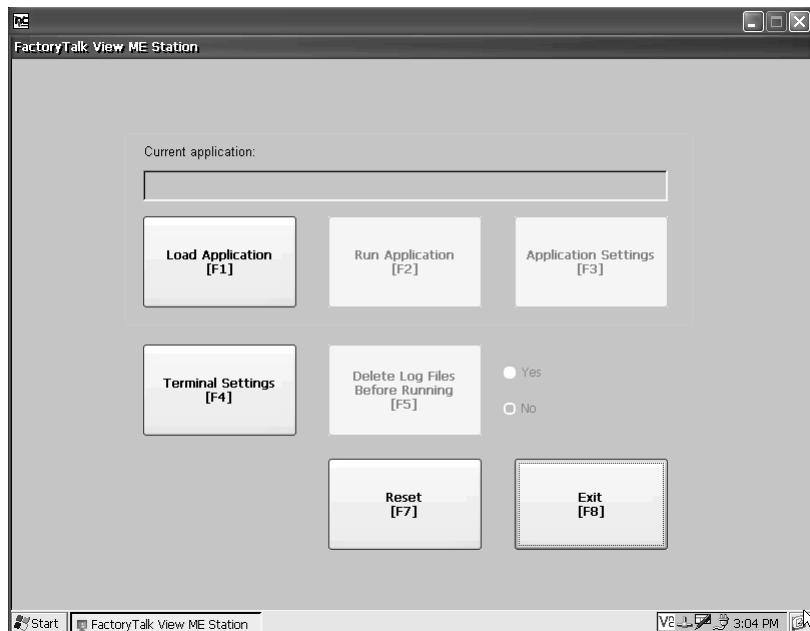
You can also transfer applications to the terminal using a Flash Card. For information about this method, see the *PanelView Plus Terminals User Manual*.

## Starting FactoryTalk View ME Station

### To start FactoryTalk View ME Station on a PanelView Plus CE terminal

1. On the Windows Start menu, select Programs, Rockwell Software, and then click FactoryTalk View ME Station.

The FactoryTalk View ME Station dialog box opens.



### Starting FactoryTalk View ME Station on a PanelView Plus terminal

If you are running a PanelView Plus terminal, the FactoryTalk View ME Station dialog box opens automatically when the terminal starts up.

### Specifying the driver to use for the transfer

You can download the runtime application file to the PanelView Plus or PanelView Plus CE terminal using:

- Ethernet
- a direct serial connection

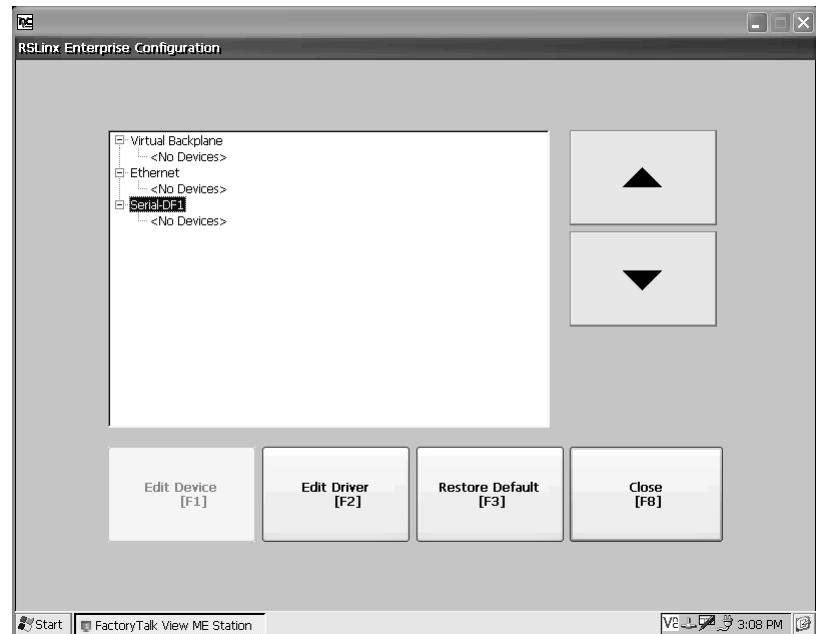
If you are using an Ethernet connection for the transfer, you don't need to set up a driver for the transfer, since the Ethernet driver is selected and loaded by default.

If you are using a serial connection for the transfer, specify and set up the driver.

You can also transfer applications to the terminal using a Flash Card. For information about this method, see the *PanelView Plus Terminals User Manual*.

### To specify and set up a serial driver for the transfer

1. In the FactoryTalk View ME Station dialog box, click Terminal Settings.
2. Double-click Networks and Communications.
3. In the Networks and Communications dialog box, double-click RSLinx Enterprise Communications.



4. Click the Serial-DF1 driver, and then click Edit Driver.
5. Select Use Auto-Configuration, and then click Edit.
6. In the dialog box that opens, select Yes, and then click OK.

Auto configuration works with most devices, including Logix5000 and PLC-5®. If auto configuration is not successful for your device, return to these steps, select No, and continue to step 7.

7. Set up the driver, and then click OK. If you are using auto configuration, skip this step.

## Setting up a driver for the transfer on the development computer

On the development computer, set up the RSLinx Enterprise driver that you will use to download your application to the PanelView Plus or PanelView Plus CE terminal.

Use one of these drivers for the transfer:

- Serial-DF1—Use for a serial transfer.
- Ethernet—Use with an Ethernet connection.

For information about setting up drivers in RSLinx Enterprise, see RSLinx Help.

## Downloading applications and Windows TrueType fonts

Use the Transfer Utility to download your runtime application and Windows TrueType font files from the development computer to the PanelView Plus or PanelView Plus CE terminal.

Runtime applications have the extension .mer. Windows TrueType font files have the extension .ttf or .ttc.

For information about creating the runtime application file, see Chapter 14.

## About the download

You can use a serial or Ethernet connection between the development computer and the PanelView Plus or PanelView Plus CE terminal destination. You must use one of the drivers listed in the previous section for the download.

You can download while an application is running on the runtime computer.

You can download:

- multiple applications (one at a time) to the same runtime computer.
- a new copy of the application currently running on the runtime computer, for use the next time the application is started.



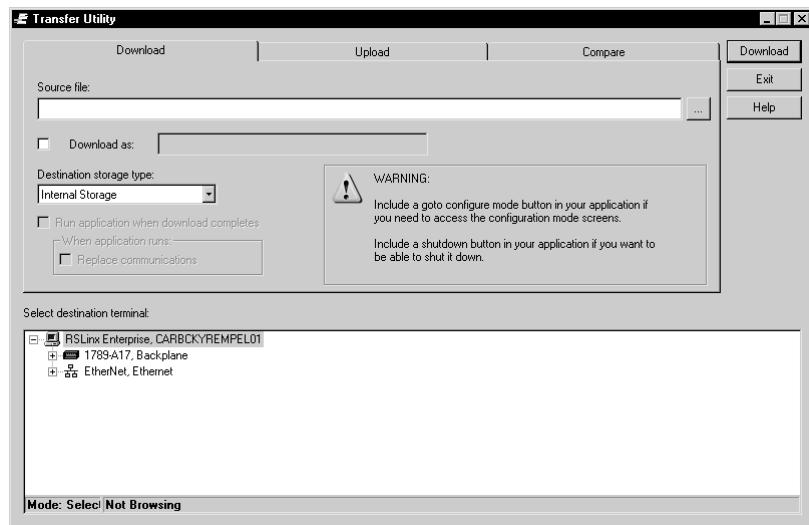
If the download process fails or is cancelled, the application file will be deleted from the destination directories. Make a copy of the application file before you begin and make sure there is enough storage space on the destination computer.

## Serial downloads

To perform a serial download, make sure the terminal is connected to the development computer using the correct cable. Connect a PanelView Plus or PanelView Plus CE terminal to the PC using the Allen-Bradley® serial cable 2711-NC13.

### To start the Transfer Utility tool

1. On the development computer, do one of the following:
  - In FactoryTalk View Studio, on the Tools menu, click Transfer Utility.
  - On the Windows Start menu, select Programs, Rockwell Software, FactoryTalk View, Tools, and then click ME Transfer Utility.



For details about using the utility, see the utility's Help.

### Uploading applications from the PanelView Plus or PanelView Plus CE terminal

Use the Transfer Utility to upload your runtime application from the PanelView Plus or PanelView Plus CE terminal to the development computer. Runtime applications have the extension .mer.

Any password changes that were made while the application was running are saved in the .mer file and will be uploaded.



If the upload process fails or is cancelled, the application file will be deleted from the destination directory. Make sure there is enough storage space on the destination computer.

## About the upload

You can upload while an application is running on the runtime computer. You can upload an existing copy of the application currently running on the runtime computer.

## Serial uploads

To perform a serial upload, make sure the terminal is connected to the development computer using the correct cable. Connect a PanelView Plus or PanelView Plus CE terminal to the PC using the Allen-Bradley serial cable 2711-NC13.

For details about performing the upload, see Help for the Transfer Utility.

## Comparing applications

You can also use the Transfer Utility tool to compare an application on the development computer with an application on the terminal. The comparison tool can tell you whether the files are identical or different.

For details about comparing applications, see Help for the Transfer Utility.



# 17 Using your application

This chapter describes:

- logging in and logging out.
- changing passwords.
- entering numeric and string values.
- changing tag values.
- viewing tag data.
- viewing alarms and messages.
- viewing information about runtime communication errors.
- changing languages.

For information about navigating between displays, see Chapter 13. For information about navigating between and using the graphic objects in a display, see page 21-3.

## Logging in to a Windows domain

To log data to a network location, the runtime computer must be logged in to the same domain as the computer on the network. To do this, the user must have access rights for the domain.

### To log in to a domain from a personal computer

1. When Windows® starts, type a user name, password, and domain name.

The PanelView™ Plus or PanelView™ Plus CE terminals cannot be part of a domain. However, you can verify that the user who is logged in to the PanelView Plus or PanelView Plus CE terminal is on a user list that is part of a domain.

On the PanelView Plus or PanelView Plus CE terminal, logging in hard codes a domain member's credentials. This gives the terminal access to permissible network resources such as network folders.

### To authenticate a user on a PanelView Plus or PanelView Plus CE terminal

1. In the FactoryTalk® View ME Station dialog box, click Terminal Settings.
2. Double-click Networks and Communications.
3. Double-click Network Connections.

4. Double-click Network Identification.
5. Type a user name, password, and domain name, and then click OK.

## Logging in to the application

Users log in using the Login window. They must have a user account in order to log in.

For information about setting up user accounts and passwords, see Chapter 11.

## 4.00 and later applications

To log in, the user must have an account in the Runtime Security editor, in the ME Runtime 4.00 and later tab. The user must have a password, which is set up in FactoryTalk Security, using the Users folder. For details, see Chapter 11.

Passwords are case sensitive.

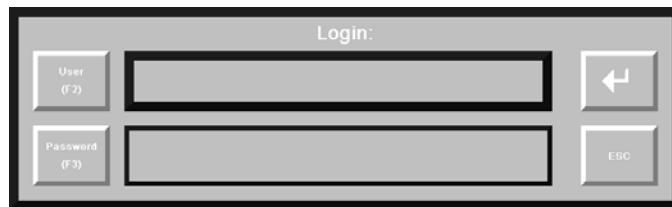
## 3.20 and earlier applications

To log in, the user must have an account in the Runtime Security editor, in the ME Runtime 3.20 and earlier tab. User passwords are optional in RSView® 3.20 and earlier. If you use them, they are not case sensitive.

### To log in

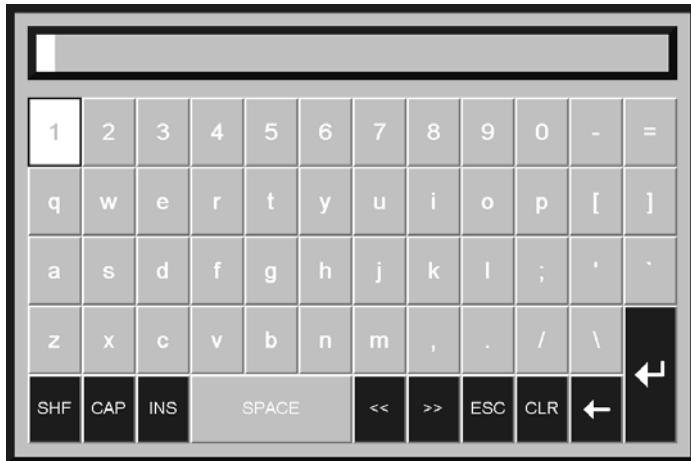
1. Press the login button.

The Login window opens.



2. Press the User button, or press F2 on an external keyboard.

The string pop-up keyboard opens. For details about using the keyboard, see page 17-11.



3. Type your user name in the string pop-up keyboard, or on an external keyboard, and then press Enter.

For FactoryTalk View 5.00 applications and RSView 4.00 applications, the name is not case sensitive. For RSView 3.20 and earlier applications, the name must use ALL CAPS.

4. To enter a password, press the Password button, or press F3 on an external keyboard.
5. Type the password in the string pop-up keyboard, or on an external keyboard, and then press Enter.

For FactoryTalk View 5.00 applications and RSView 4.00 applications, the password is case sensitive. For RSView 3.20 and earlier applications, the password is not case sensitive.

6. To close the Login window and complete the login, press Enter.

## What happens when a user logs in

When a user logs in:

- If the previous user did not log out, the previous user is logged out now. If a logout macro is set up for the previous user, the logout macro runs, assigning values to tags. If the previous user belongs to a group, and a logout macro is set up for the group, the logout macro runs.
- The new user is logged in.

- If a login macro is set up for the new user, the macro runs, assigning values to tags. If the new user belongs to a group, and a login macro is set up for the group, the login macro runs.

## Problems with logging in

Login is unsuccessful under these circumstances:

- If the graphic display changes remotely before the user has finished logging in, the login is cancelled.
- If the user name doesn't match the settings in the Runtime Security editor, the user is not logged in.
- If the user password doesn't match the password set up in FactoryTalk Security, in the Users folder (4.00 or 5.00 applications), or the password set up in the Runtime Security editor (for 3.20 and earlier applications), the user is not logged in.
- If the screen resolution is smaller than 280 pixels wide by 84 pixels high, the Login window cannot open.
- If the screen resolution is smaller than 236 pixels wide by 208 pixels high, the string pop-up keyboard cannot open.

## Changing passwords

You can use the password button to change your password at run time. Any password changes that are made at run time are saved in the .mer file.

When passwords are changed at run time, be sure to change the passwords on the development computer as well. Otherwise, the next time you create the .mer file the password changes will be lost. For version 5.00 applications, you can convert the runtime application file to a development application in order to save password changes. For more information, see page 14-6.



If you are logged in to Windows as an Administrator user, you can use the password button to change passwords for Windows-linked users. If you are not logged in as an Administrator user, or if your application is running in Windows CE, the passwords for Windows-linked users must be changed in Windows.

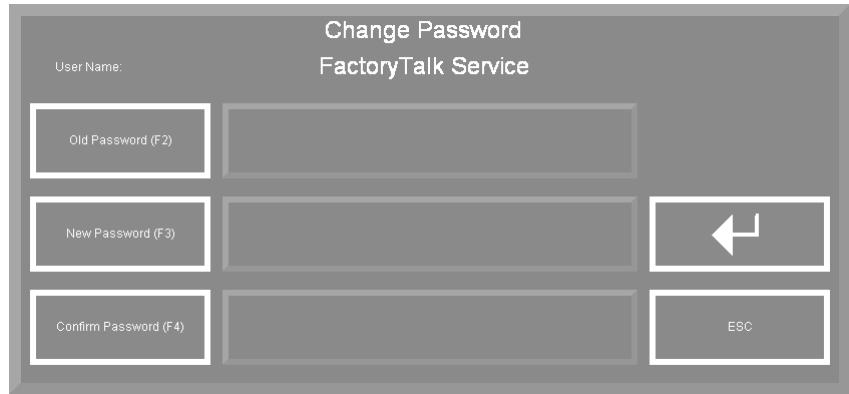


For version 3.20 and earlier applications, you cannot change passwords in FactoryTalk View ME Station. To change a password for a FactoryTalk View user, use the Runtime Security editor in FactoryTalk View Studio, then recreate the runtime application and download the .mer file again. To change a password for a Windows user, use the Windows Control Panel.

### To change your password

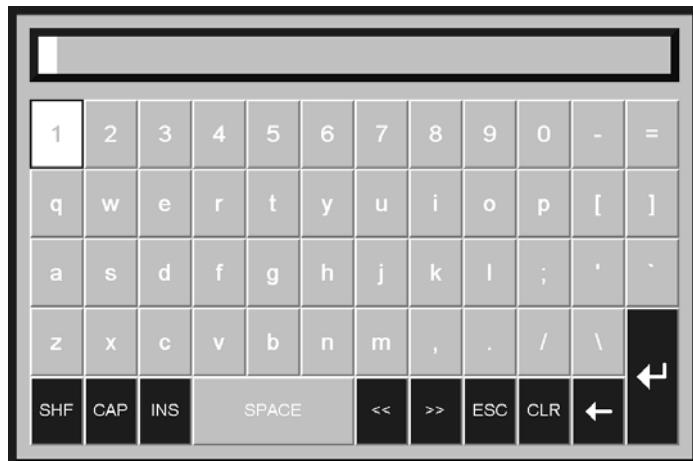
1. Press the password button.

The Change Password window opens.



2. Press the Old Password button, or press F2 on an external keyboard.

The string pop-up keyboard opens. For details about using the keyboard, see page 17-11.



3. Type your old password in the string pop-up keyboard, or on an external keyboard, and then press Enter.

The password is case sensitive.

4. Press the New Password button, or press F3 on an external keyboard.

5. Type the new password in the string pop-up keyboard, or on an external keyboard, and then press Enter.
6. Press the Confirm Password button, or press F4 on an external keyboard.
7. Type the new password again, and then press Enter.
8. To close the Change Password window and complete the change, press Enter.

## Logging out

When the current user logs out, if a logout macro is assigned to the user, the macro runs. If the user belongs to a group, and a logout macro is set up for the group, the logout macro runs.

After the current user is logged out, the DEFAULT user is logged in. If a login macro is assigned to the DEFAULT user, the macro runs.

You can also set up the application to automatically log out the current user after a specified period of inactivity. For more information, see page 11-13.

### To log out

1. Press the logout button.

## Entering numeric values

The operator can enter or ramp numeric values at run time using the numeric input enable button or the numeric input cursor point.

When the operator presses the button or activates the cursor point, the numeric pop-up keypad or scratchpad opens. If you set up the button or cursor point to ramp, pressing the object gives it focus, but does not open a pop-up window.

To use the numeric pop-up scratchpad, a keyboard must be attached to the runtime computer, or the terminal must be a keypad terminal.

For information about setting up the numeric input enable button and numeric input cursor point, see Help.

You can also use a ramp button to ramp numeric values. For details, see Help for the button.

### Activating the cursor point

When the numeric input cursor point has focus, the operator can activate the cursor point by pressing any of these keys on the keyboard or keypad:

- numbers from 0 to 9

- the minus sign ( - ) or decimal point ( . )
- the Enter key, or an enter button
- the Backspace key, or a backspace button

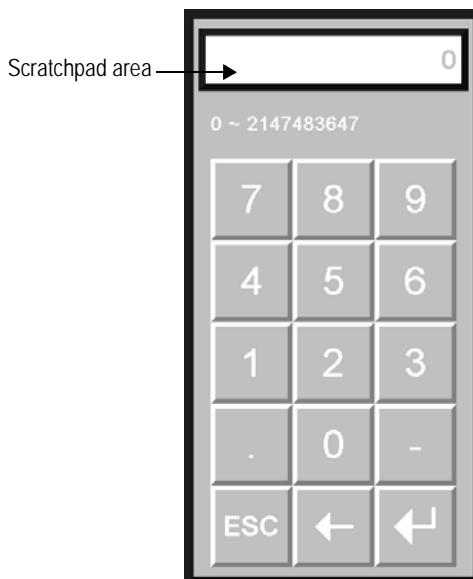
When the operator activates the numeric input cursor point, the numeric pop-up keypad or scratchpad opens.

### Ramping numeric values

If you set up the numeric input enable button or numeric input cursor point to ramp values, the operator presses the object to give it focus. When the object has focus, the operator can press a move up or move down button, or the Up Arrow or Down Arrow key on a keyboard or keypad, to ramp the value at the Value connection.

### Using the numeric pop-up keypad

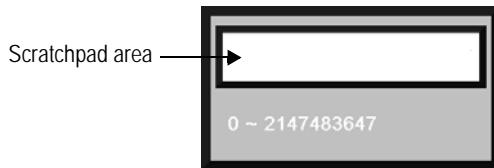
The keypad can accept up to 17 digits, including the decimal point, leading zero, and minus sign.



When the keypad is open, no other objects in the graphic display can accept input.

### Using the numeric pop-up scratchpad

The scratchpad can accept up to 17 digits, including the decimal point, leading zero, and minus sign.



The scratchpad does not contain any buttons. If the runtime computer does not have an external keyboard or keypad attached, the operator will be unable to enter values or close the scratchpad.

When the scratchpad is open, no other objects in the graphic display can accept input.

### Using buttons and keys with the numeric pop-up windows

You can use the following buttons and their keyboard or keypad equivalents with the numeric pop-up keypad. For the pop-up scratchpad, you can use the keys on an external keyboard or keypad only.

Use this button or key	To do this
Decimal (.)	Type a decimal point.  This button is visible only if the decimal point type for the numeric input enable button or numeric input cursor point is Implicit.
Minus (-)	Toggle the value between positive and negative.
ESC	Close the pop-up window without writing the value to the object's Value connection.
Backspace	Delete the right-most digit in the scratchpad. If no digits are left, the minus sign, if any, is removed.
Enter	Close the pop-up window and write the value to the object's Value connection.  If the Optional Expression connection is assigned to the button or cursor point, FactoryTalk View calculates the value of the expression and writes the result to the Value connection. For details, see page 23-19.  If Enter key handshaking is set up for the button or cursor point, the object's Enter connection is set to 1 and the handshaking timer begins timing. For details, see page 21-13.
Delete	Clear the scratchpad.  There is no Delete button in the pop-up keypad, but you can use the Delete key on an external keyboard or keypad.

## How values are ramped

If the button or cursor point is being used to ramp a value at the Value connection:

- When the tag assigned to the Value connection is an integer tag, but the ramp value is a floating point value, the ramp value is added to (or subtracted from) the Value connection value, and then the result is rounded and written to the Value connection.

For information about how values are rounded, see page 7-2.

- If the operator presses and holds down the move up or move down button, or the Up Arrow or Down Arrow key on the keyboard or keypad, the button or key goes into auto repeat mode. The ramp value is added to (or subtracted from) the last value sent to the Value connection.

For more information about auto repeat mode, see page 21-13.

- If ramping the button or cursor point would cause the value at the Value connection to exceed the maximum value, or be less than the minimum value, a message is written to the Diagnostics List and the value at the Value connection is not changed.

## How values are calculated

When the pop-up keypad or scratchpad has focus and the operator presses the Enter button, the value that is sent to the Value connection depends on how the button or cursor point is set up.

- The value the operator enters is compared to the minimum and maximum range for the object. If the value is within the range, the value is written to the Value connection.

If the Optional Expression connection is assigned, and the original value is within the range but the expression result is a value outside the range, the value is still written to the Value connection.

- If the decimal point type is Fixed Position, with a “Digits after decimal” value that is greater than 0, the decimal point is stripped from the value before comparing it to the minimum and maximum range.

For example, if the entered value is 9.25, the stripped value is 925.

If the stripped value is within the range, the stripped value is sent to the Value connection (or substituted for the ? in the Optional Expression).

- If the decimal point type is Implicit, and the tag assigned to the Value connection uses an integer data type, floating-point values are rounded.

If the Optional Expression connection is assigned, the floating-point value is substituted for the ? in the expression, and the expression result is rounded.

For information about how values are rounded, see page 7-2.

## Problems with the numeric pop-up windows

Problems with the numeric pop-up keypad and scratchpad occur under these circumstances:

- If the graphic display changes remotely before the user has pressed the Enter button, the pop-up window closes without writing out a value.
- If the screen resolution is smaller than 124 pixels wide by 240 pixels high, the pop-up keypad cannot open.
- If the screen resolution is smaller than 124 pixels wide by 68 pixels high, the pop-up scratchpad cannot open.
- If the value is too large for the data type of the tag assigned to the Value connection, the pop-up window remains open and the value is not written to the connection. The scratchpad area changes to red to alert the operator of the error.
- If the value is outside the minimum and maximum range specified for the object, the pop-up window remains open and the value is not written to the connection. The scratchpad area changes to red to alert the operator of the error.
- If Enter key handshaking is still in effect, the pop-up window closes but the value is not written to the connection.

## Entering string values

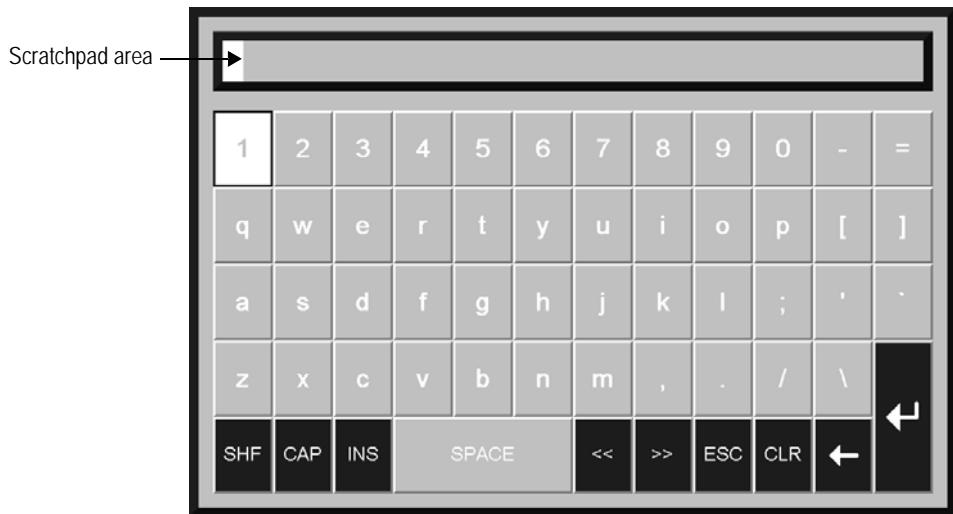
The operator can enter string values at run time using the string input enable button.

When the operator presses the button, the string pop-up keyboard or scratchpad opens, depending on how you set up the button. To use the scratchpad, a keyboard must be attached to the runtime computer.

For information about setting up the string input enable button, see Help.

## Using the string pop-up keyboard

The string pop-up keyboard opens when the operator presses a string input enable button or the User or Password button in the Login dialog box. The pop-up is also used with the password button graphic object.



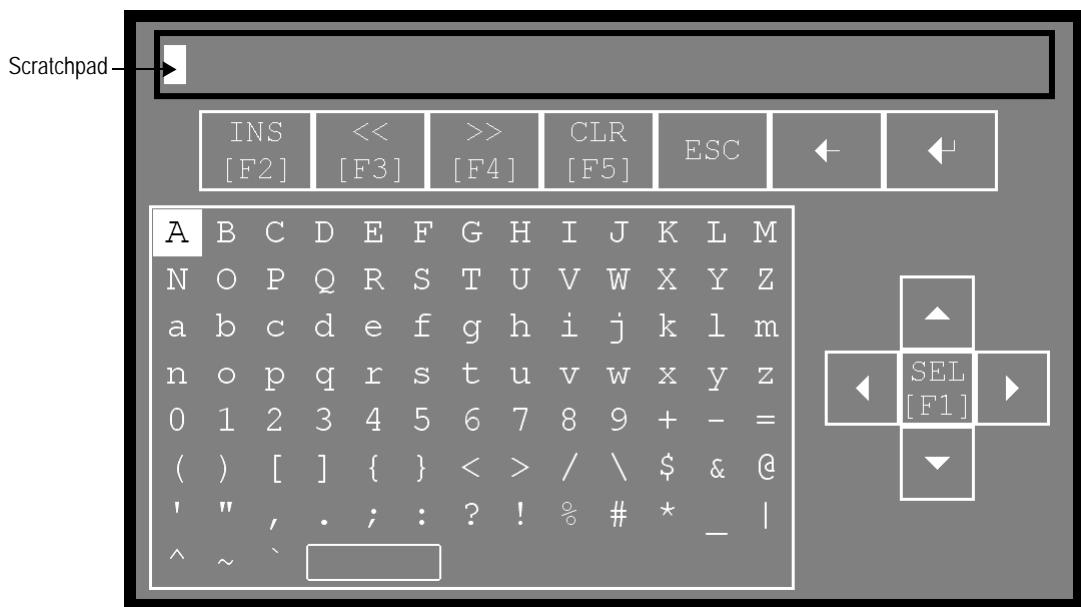
If the operator opens the pop-up keyboard by pressing a string input enable button, the number of characters the keyboard accepts depends on how you set up the button.

When the keyboard is open, no other objects in the graphic display can accept input.

## Using the string pop-up character input

You can use a string pop-up character input instead of the string pop-up keyboard. With the string pop-up character input, you use the arrow keys to select the characters to input.

The string pop-up character input opens when the operator presses a string input enable button or the User or Password button in the Login dialog box. The pop-up is also used with the password button graphic object.



If the operator opens the string pop-up character input by pressing a string input enable button, the number of characters the input accepts depends on how you set up the button.

### To use the string pop-up character input on a personal computer

1. In FactoryTalk View Studio, on the Tools menu, select Options.
2. Click the String Pop-Up tab.
3. Select Use the string pop-up character input.

### To use the string pop-up character input on a PanelView Plus or PanelView Plus CE terminal

1. In FactoryTalk View ME Station, click Terminal Settings.
2. Double-click Input Devices.
3. Select String Pop-Up, and then click Enter.

4. Specify whether to invoke the pop-up character input instead of the pop-up keyboard, and then click OK.

## Using the string pop-up scratchpad

If the operator opens the string pop-up scratchpad by pressing a string input enable button, the number of characters the scratchpad accepts depends on how you set up the string input enable button.



The scratchpad does not contain any buttons. If the runtime computer does not have an external keyboard attached, the operator will be unable to enter characters or close the scratchpad (unless the runtime terminal is a keypad terminal). If the runtime terminal is a keypad terminal, the operator can enter numbers (not letters) in the scratchpad, and close the scratchpad.

When the scratchpad is open, no other objects in the graphic display can accept input.

## Using buttons and keys with the string pop-up windows

You can use the following buttons and their keyboard equivalents with the string pop-up keyboard. For the pop-up scratchpad, you can use the keys on an external keyboard only. The string pop-up character input doesn't have a SHF or CAP key, and it has additional arrow keys for selecting the characters to input.

Use this pop-up keyboard button	Or this keyboard equivalent	To do this
SHF	none	Capitalize a single letter, or type a shifted character such as #.
CAP	none	Capitalize multiple letters.
INS	Insert	Toggle between insert and overstrike modes.
SPACE	Spacebar	Insert a space.
<<	Left Arrow	Move the cursor to the left.
>>	Right Arrow	Move the cursor to the right.

Use this pop-up keyboard button	Or this keyboard equivalent	To do this
ESC	Esc	Close the pop-up window without writing the string to the Login dialog box, the Change Password dialog box, or the string input enable button's Value connection.
CLR	Delete	Clear the scratchpad.
Backspace	Backspace	Delete the character in front of the cursor.
Enter	Enter	<p>Close the pop-up window and write the string to the Login dialog box, the Change Password dialog box, or the string input enable button's Value connection.</p> <p>If Enter key handshaking is set up for the string input enable button, the button's Enter connection is set to 1 and the handshaking timer begins timing. For details, see page 21-13.</p>

## What is written to the Value connection

When the operator presses the Enter button in the pop-up keyboard or scratchpad, the string that is sent to the Value connection depends on how the string input enable button is set up.

- If a fill character is set up for the button, and the operator enters fewer than the maximum number of input characters, the fill characters are placed after the string the operator enters.
  - Spaces have a hex value of 20.
  - Zeroes have a hex value of 30.
  - FF characters have a hex value of FF.
  - Null characters have a hex value of 0. The null character indicates the end of string input. It does not add to the actual string length.
- If the number of input characters is fewer than the number of characters in the length of the string tag assigned to the Value connection, the remaining spaces are padded with the null character.

When the string is written to the Value connection, the first character is placed in the high order byte of the first word at the tag address, the second character is placed in the low order byte of the first word, and so on.

## Problems with the string pop-up windows

Problems with the string pop-up keyboard and scratchpad occur under these circumstances:

- If the graphic display changes remotely before the user has pressed the Enter button, the pop-up window closes without writing out a string.
- If the screen resolution is smaller than 236 pixels wide by 208 pixels high, the pop-up keyboard cannot open.
- If the screen resolution is smaller than 236 pixels wide by 44 pixels high, the pop-up scratchpad cannot open.
- If the string pop-up window is set up to accept more characters than the Value connection tag length, the pop-up window remains open and the string is not written to the connection. The scratchpad area changes to red to alert the operator of the error.
- If Enter key handshaking is still in effect, the pop-up window closes but the value is not written to the connection.

## Changing tag values

This section gives an overview of the graphic objects you can use to change tag values. For information about setting up the objects, see Chapter 21 and Help.

The operator uses these objects to start and stop plant operations, and to control machines and processes. Choose the objects that best suit your process. Set up the data source to carry out the desired actions in response to the changes in tag values.

The operator can use function keys with all of these objects except control list selectors, third-party ActiveX® objects, and drawing objects with slider animation. The last three types of objects can be pressed using a mouse or touch screen.

For information about assigning function keys to graphic objects, see page 21-5.



Do not use push buttons for emergency stops. Emergency stop buttons must be hard-wired.

You can also use macros to assign values to tags. For more information, see Chapter 30.

The operator can change tag values at run time using these graphic objects:

### Use this graphic object

### To do this

---

Momentary push button

Start a process or action by sending one value to the tag when pressed, and another value when released.

---

Use this graphic object	To do this
Maintained push button	<p>Toggle between two values by sending one value to the tag when pressed, and a second value the next time the button is pressed and released.</p> <p>This button is useful for changing a setting within a machine or process, but not for starting the machine or process.</p>
Latched push button	<p>Start a machine or process and remain set (latched) until the process is completed, by sending a value to the tag when pressed, and retaining this value until reset (unlatched) by the Handshake connection.</p>
Multistate push button	<p>Cycle through a series of values. Each time the operator presses the button, the value for the next state is sent to the tag. When the button is in its last state, pressing it causes the button to change to its first state and write out the first state value.</p> <p>This button is useful when you want the operator to see and select multiple options in sequence, using a single button. The button displays the current state of a process or operation by showing a different color, caption, or image to reflect the different states.</p>
Interlocked push button	<p>Use a group of buttons to send values to the same tag. When the operator presses a button in the group, the button's value is sent to the tag, and the button remains highlighted as long as the tag value is the same as the button's value. Pressing a new button in the group releases the other button and sends a new value to the tag.</p> <p>You can also use a single interlocked push button to send a value to a tag.</p>
Ramp button	<p>Increase or decrease the value of a tag by a specified integer or floating-point value.</p> <p>Use two ramp buttons together to create a raise/lower control.</p>
Numeric input enable button	<p>Enter a numeric value and write the value to a tag. You can also use this object to ramp values.</p>
Numeric input cursor point	<p>Enter a numeric value and write the value to a tag. You can also use this object to ramp values.</p>
String input enable button	<p>Enter a string value and write the value to a tag.</p>
RecipePlus button	<p>Write values for all the ingredients in the selected recipe to a set of tags. The button works with the RecipePlus table and RecipePlus selector graphic objects.</p>



Use this graphic object	To do this
Drawing object with horizontal or vertical slider animation	Control the value of a tag by dragging the slider object with a mouse. The pixel position of the slider is translated into a value that is written to the tag.  If the value of the tag is changed externally, the position of the slider changes to reflect this.
Control list selector	Select from a list of states for a process or operation. The list is highlighted to show the current state, and the operator can scroll through the list to select a different state. The value assigned to the selected state is written to the tag.  If the value of the tag is changed externally, the position of the highlight changes to reflect this.
ActiveX object	A third-party object, connected to an analog, digital, or string tag, including both HMI and data server tags. When the object's property value changes, the new value is written to the associated tag.

You can attach visibility animation to these graphic objects, to display or hide the objects based on changes in tag or expression values. For information about visibility animation, see page 22-8.

For information about creating graphic objects, see Chapter 20. For information about setting up graphic objects, see Chapter 21 and Help.

## Viewing tag data

This section gives an overview of the graphic objects you can use to display tag data. For information about setting up the objects, see Chapter 21 and Help.

The operator can view tag data at run time using these graphic objects:

Use this graphic object	To display this
Numeric display	Numeric tag values. For example, display the current temperature of an oven.
Numeric input cursor point	Numeric tag values. For example, display the current temperature of an oven.
String display	String tag values. For example, set up the data source to generate strings that report on the state of a process or operation, or that provide the operator with instructions about what to do next.

Use this graphic object	To display this
Bar graph	Numeric values in bar graph format. The bar graph increases or decreases in size to show the changing value.
Gauge	Numeric values in dial format. The gauge's needle moves around the dial to show the changing value.
Multistate indicator	The state of a process, on a panel that changes its color, image, or caption to indicate the current state. Each state is set up to correspond to a numeric tag value or least significant bit.
Symbol	The state of a process, using a monochrome image that changes color to indicate the current state. Each state is set up to correspond to a numeric tag value or least significant bit.
	This object is useful for showing the state of a process or operation at a glance.
List indicator	The state of a process, using a list of possible states with the current state highlighted. Each state is represented by a caption in the list, and is set up to correspond to a numeric tag value or least significant bit.
	This indicator is useful if you want to view the current state but also want to see the other possible states. For sequential processes, the list can alert the operator about what happens next in the process.
Trend	Historical or current numeric tag values, plotted against time or displayed in an XY plot where one or more tags' values are plotted against another tag's values to show the relationship between them.
RecipePlus table	Current tag values and data set values of the ingredients in the selected recipe, and the number of ingredients in the recipe. The table works with the RecipePlus button and RecipePlus selector graphic objects.
Drawing object with rotation, width, height, fill, color, or horizontal or vertical position animation	Display the value of a tag using a pictorial representation that shows the current value in relation to a range of possible values. For example, use rotation animation to show the tag value as a needle's position on a dial.
	For color animation, assign different colors to represent different values.
ActiveX object	A third-party object, connected to an analog, digital, or string tag, including both HMI and data server tags. The data displayed depends on the object.

Many of these objects can be set up to manipulate tag values using expressions, and display the expression result rather than the original tag value. For information about expressions, see Chapter 23.

You can also attach visibility animation to these graphic objects, to display or hide the objects based on changes in tag or expression values. For information about visibility animation, see page 22-8.

For information about creating graphic objects, see Chapter 20. For information about setting up graphic objects, see Chapter 21 or Help.

### Displaying the date and time

To display the current date and time, create a time and date display. This object uses the operating system's date and time, in the format of the application's current language, and therefore does not require tags or expressions.

## Viewing alarms and messages

The operator can view alarms and other messages at run time using these graphic objects and graphic displays:

This information	Appears in this object	In this default graphic display	For details, see
Alarm messages	Alarm banner	[ALARM].	page 9-26
Alarm messages	Alarm list	No default, although this object appears in the [ALARM MULTI-LINE] and [HISTORY] graphic libraries.	page 9-25, page 9-27, and page 9-29
Alarm messages	Alarm status list	No default, although this object appears in the [STATUS] graphic library.	page 9-28
System activity	Diagnostics list	[DIAGNOSTICS]	page 10-11
Information messages	Information message display	[INFORMATION]	page 27-7
Local messages	Local message display	No default.	page 19-29

The default alarm and information displays open automatically when the assigned tags match messages' trigger values. The default diagnostics display opens automatically when system activity occurs. If desired, you can set up your own graphic displays to open automatically, instead of the default displays. You can also set up any of the displays to open when an operator presses a goto display button or selects a display in the display list selector.

The operator can acknowledge alarm and information messages. The operator can clear alarm and diagnostics messages. The operator can sort alarms and reset their status.

## **Viewing information about runtime communication errors**

To display communication errors in the diagnostics list object, set up message routing so that messages are sent to the FactoryTalk View Diagnostics List.

For information about setting up diagnostics message routing, see Chapter 10.

## **Changing languages**

You can change languages at run time. The languages available depend on what has been set up for the runtime application. There is a separate language switch button for each language that you can change to.

For information about setting up language switching, see Chapter 12.

### **To change languages**

1. Press a language switch button.

Text strings in the application change to the language specified by the button.